ETHERISC

White Paper 2.0

DRAFT - 26.11.2022

Table of Contents

Acknowledgements

The ideas in this paper are a collaborative result of many talks and discussions, online and in person, with some brilliant minds of the crypto-economic space. The Etherisc team would like to say “thank you” especially to the following people, who added considerable value to the draft: Ron Bernstein, Jake Brukhman, Alexander Bulkin, Alex Felix, William Mougayar, Micah Zoltu, to the current team, consisting of Matthias Zimmermann, Jan Stockhausen, Michiel Berende, Hui Lin Chiew, Sebastian Schlitter, Peter Koch, Koenraad de Jonghe, Damian Groß, Felizitas Mussenbrock-Strauß, Lydia Mussenbrock and last but not least all those members of our telegram channel and discord server, who gave valuable feedback and criticism and encouraged us to follow our path.

1 About this Document

History

As a result of the Nov/Dec 2016 Hackathon, the Etherisc team wrote a White Paper, which was released to the public in its Version 0.3. Focus of the hackathon was the attempt to outline the core of a blockchain based reinsurance market.

Two years later, we released version 1.01 of the whitepaper for the DIP tokensale which took place from June 23rd to July 25th, 2018. The 2018 whitepaper outlined the main features of the Etherisc ecosystem, and also important aspects of the DIP token. It was already clear that staking would emerge as the core token utility, however, a long way had to be gone to iron out the details.
We also had only a rough idea of what the insurance framework would look like. Immediately after the token sale, we started implementing the framework which was released as “GIF Framework” in 2019.

The Elevator pitch

Giant corporations dominate the multi-trillion-dollar insurance industry. The insurance industry developed into an indispensable part of the modern economy, whilst sometimes being inefficient, intransparent, expensive and with inferior customer experience. It has become a commonplace to say that insurance is boring and sucks. When customers need help the most, they can struggle in vain to get reimbursed by companies whose profits too often depend on not paying.

Etherisc has built a platform for decentralized insurance applications that anyone can use. Large and small companies, non-profit groups, or insurtech startups.

The Etherisc platform is built around a technical, blockchain-based Framework called “GIF”. GIF is an acronym that stands for “Generic Insurance Framework”. It consists of open-source smart contracts that implement generic insurance product and policy lifecycle functions. Thus, GIF enables a wide range of insurance types. GIF runs on a blockchain and is multi-chain and multi-tenant capable.

The GIF is particularly suited to host parametric insurance products. Parametric insurance covers predefined loss events immediately when they occur, with predefined and deterministic payouts, rather than compensating for actual damage. Events such as flight delays, droughts, heavy rainfall, or damage caused by hurricanes are covered.

In parametric insurance loss events (the risks) are defined as functions of underlying indices or parameters that meet the criteria defined by the insurance product.

The GIF is open source, so anybody can deploy his own GIF instance, modify the code, fork it etc. However, we believe that a flourishing platform needs more that just some lines of code.
Therefore, any GIF instance can optionally be registered in a global registry which is maintained by a DAO, in which all platform stakeholders participate in the governance. Registered GIF instances need to comply with a set of rules which ensure that customers can feel safe when they interact with a registered GIF instance. This is achieved by granting badges and certificates to compliant GIF instances.

In this system, the native DIP token (Decentralized Insurance Protocol) plays a central role. Every participant and stakeholder is required to stake a certain amount of DIP tokens. Therefore, the DIP token is a utility token, with staking being the central utility. Staking is needed both for deploying products, oracles and risk pools, but also for investing in those risk pools. Staking will also grant voting rights in the governance model.

We aim to develop the Etherisc platform to an open, growing ecosystem which is owned and governed by the community of all participants, creating insurance products which are transparent, fair and accessible for both customers and investors.

Chapters

This White Paper is the attempt to structure the overall picture of decentralized insurance along the new insights. The document is structured in 4 main chapters:

In chapter two, we explain the fundamental facts about insurance and the disadvantages of traditional insurance. We highlight the advantages of blockchain and the decentralized model in insurance and explain the interplay between customers, users and companies.

Chapter three explains the Etherisc perspective on insurance, policies, and parametric insurance. We define the three pillars of the Etherisc ecosystem and get into the technical side of GIF (Generic Insurance Framework). We define the generic lifecycle of the GIF functions. At the end of the chapter, we introduce the GIF Monitor.

Chapter four explains the DIP token and how it is used in Etherisc staking. We describe our staking model, the interaction of the different corresponding risk pools and the functions of the different tokens. The chapter is concluded with the topic of credits rewards and payment losses.

Chapter five deals with EGM (Etherisc Governance Model). We define the five core values as respect, partnership, responsibility, trust and public goods. They were followed by the structure of EGM and the participants and their roles. Details about the topics of global staking pool and monetary policy and then round off the white paper with the appendix.

The final **chapter six** contains a list of all abbreviations and definitions of the most important terms.

Ggg

2 Analysis of Basic Insurance Paradigms

2.1 Overview

To understand the approach, strategies and goals of (this) insurance, it's important to look at a general definition of what insurance is and does:
"Insurance is a means of protection from financial loss in which, in exchange for a fee called "premium", a party agrees to guarantee another party compensation in the event of a certain loss, damage, or injury. It is a form of risk management, primarily used to hedge against the risk of a contingent or uncertain loss."[1] This can occur as a contract in the form of a financial protection policy.

The insured is the policyholder whereas the insurer is the insurance - providing company/ the insurance carrier/ the underwriter.[2]

Having analyzed the basic principles of insurance, such as the definition, and having developed a token system on top of these principles, we can now analyze insurance and break costs and capital flows down into three elements:

Expected value of risk
Which is simply the redistribution of capital according to the risks among the participants.

Capital costs for long-tail risks
As the name indicates, the capital in risk pools is exposed to a risk of loss and must be held in the risk pool for a certain period. The capital providers are compensated for this risk. This compensation is calculated based on the limitation period and the insured risk.

Transaction costs
Costs of administration of insurance policies, for example, gas fees of bookings on the blockchain, booking fees in case of payment in FIAT money, costs for oracles, etc.

We argue that traditional insurance companies dominate these building blocks and thus rule the insurance market. The GIF framework and the underlying blockchain technology offer the opportunity to replace the encrusted processes of traditional

---

[1] https://en.wikipedia.org/wiki/Insurance
[2] https://paytm.com/blog/insurance/what-is-insurance-definition-benefits-and-types/

insurance companies with lean decentralized structures using standardized and automated lean protocols. Tokens thereby map the capital and revenue flows.
Our conclusion from this analysis is that we need two types of tokens. The first one - the "DIP Token" - supports the coordination and economical incentivization of actors in a decentralized insurance system.
The second type of token represents risks - this type is not a single token but a class of similar tokens, one for each risk pool, we call those "risk pool tokens".
In a distributed environment with many participants, building products as a collaborative effort, the protocol token serves as glue, as collateral, and as representation of the material and immaterial value of the network, much as Ether serves as a means to secure the stability of the Ethereum Blockchain.

In Chapter 4.1, we detail the DIP protocol token. Chapter 6 shows a concrete example of the use of the token in an insurance context.

2.2 Principles of insurance

We explain the principle of insurance with an example . The example is of course simplified, and serves the sole purpose to explain the principle.

We consider a homeowners insurance. Insurance is about probabilities of losses, so it would be interesting to see what the probability of a damage is. A homeowners insurance typically covers a number of perils, including fire, natural disasters, water, and even falling objects[3]. But it is difficult to obtain real numbers, as insurance companies are not very transparent with their fundamental data[4]. We will assume that, for our example, the probability is 0.1%.

For our fictional example, let's assume insurance had not been invented yet. In this fictional world, Alice owns a house. The house is worth $100K. The probability of a complete disaster is 0.1% per year (that is one devastating event in 1,000 years). Alice wants to ensure that she has access to enough funds to get a new house in the case

[3] Allstate.com: What Perils Are Typically Covered By A Homeowners Insurance Policy?

[4] A quick market survey in Germany shows that you get a homeowners insurance for considerably less than 0.1% of the value. For simplicity, we'll assume that the premium is 0.1% plain and we don't take insurance taxes etc. into account.
From the relation premium/value, we can easily estimate an upper bound for the probability. One of the most fundamental principles of insurance is that the expected losses should not surpass the collected premiums ("Risk loading" - cf. http://www.wiley.com/legacy/wileychi/eoas/pdfs/TAP027-.pdf). The expected losses are - simplified - number of policies multiplied with the probability of loss multiplied with the loss (which is equal to the value), and collected premiums are number of policies multiplied with premium per policy. It follows that the probability can be approximated by premium/value, which is lower than 0.1% in our market test.

of a disaster. So she decides to get a loan of $100K and has to pay redemption (also called principal) and interest rate.
Additionally, she pays an interest rate of maybe 1%, so she has yearly costs of $1,100 ($100,000 loan * 1% interest rate plus $100 annual redemption = $1100.00).
Now we show how pooling risks in an insurance scheme reduces these costs drastically.

2.2.1 Sharing the expected value of risk

Assume 100,000 homeowners are coming together in a pool. Again, everybody pays a $100 share; this amount is now called the "premium". They collect a total of $10,000,000 in premiums. But now there is a difference to Alice, who takes care only for herself: because of the law of large numbers[5], with a very high probability there will only be about 100 fires, causing a damage of about $10,000,000! And because the sum of all premiums is also $10,000,000, the whole damage can be paid out of the collected premiums, there is no need for every house owner to take on a loan.
(Because premiums are collected at the beginning of the year, and all the houses "expected" to burn don't all burn at the beginning of the year, but more or less are equally distributed over the year(s), there is a so called "float"[6] of liquidity which can also generate a significant revenue. For simplicity, we won't focus on this effect in this paper.

So the costs for each single house owner are now reduced from $1,100 to $100!
This difference asks for an economical explanation. Let's have a closer look into it.
First of all, if all house owners would follow Alice's example, they would need a huge amount of loans, from which only a tiny part of 0.1% would been needed in the average. It is clear that providing unused liquidity is costly. Pooling of risks in an insurance optimizes the use of capital, and the participants benefit from the reduced costs, not to speak from the difficulties to obtain a loan without collateralization!
Second, if everybody only cares for himself, only a tiny fraction of participants are struck by disaster, and have the burden of actually paying back their loan. The others can pay back without loss, as soon as they don't need protection. In an insurance collective, we have solidarity: with the premiums, everybody pays for the damages of the others.
To summarize, the risk pool offers three advantages for the participants:

1. Building a large liquidity pool.
2. Guaranteed access to this liquidity in case of a damage.
3. Mutual subsidizing of damages.

Such a pool may be designed solely to benefit its' participants, and to not make any "profit". If the pool did generate profits, these profits could be distributed back to the

---

[5] https://en.wikipedia.org/wiki/Law_of_large_numbers
[6] http://www.npr.org/sections/money/2010/03/warren_buffett_explains_the_ge.html

participants, effectively reducing the premiums again to a level where no profits are generated. Such an insurance would have a loss ratio of 100%, because all premiums are used to pay the losses.

This is the very basic effect of risk transfer in insurance. Please note that the effect increases with the pool size.

But still, this is not the whole story.

2.2.2 Sharing the long-tail-risks

In some years, there are more fires, in other years, less. To account for these variations in damages, the whole pool has to raise some money, e.g. $10M, to cover the unlikely event of a burst of many fires in one particular year. And let's suppose that the interest rate for this capital is even particularly high, e.g 20%. We will have total costs for this capital of $2M. The interest rate for the capital is a function of the risk and the riskless interest rate on the capital market; in an efficient market, the interest rate will compensate for the higher risk in comparison with a risk-free investment and will also contain a fair profit. So basically, this is where profits are generated for providing capital in an insurance structure.

The overall costs of $2M are distributed among all house owners, yielding an additional cost of $20 per house owner per year, which is added to the premium.

So after this, there is also a protection against "long tail risks" or "black swan events", at a cost of $20 per house owner. Again, the risk diversification effect increases with the pool size.

Overall, participants now pay $120 per year for their house insurance. The loss ratio is now reduced to 83% because of the capital costs of protecting the long tail risks[7].

2.2.3 Sharing the transaction costs

To organize 100,000 people in a pool, a professional structure is needed, otherwise, every single participant would have to talk to every other, which would simply be impossible. The operation of this professional structure adds transaction costs to the premium. This is the reason why insurance companies have come into existence: They provide a way to decrease transaction costs for the participants of the pool, creating an economy of scale and coordinating a huge number of participants and employees[8]. The effect is considerable and enables the modern form of insurance

---

[7] $100 for covering the risk against $120 premium => 100/120 loss ratio = 83%

[8] The downside of this is the fact that inefficiencies tend to hide in the organization. The bigger the organization, the lesser people are doing the real work (people at the "rim" of the organization) and the

with huge customer bases and a capitalization which can cover even global catastrophic events like hurricanes and earthquakes. However, the remaining transaction cost are still considerable: a recent study by KPMG shows the impact on the loss ratio, which is about 66% in the average[9].

2.2.4 Information asymmetry

Together with the reduction of transaction costs comes an asymmetry of information, which leads to a further increase of costs and to incredible profits for the big insurance companies. The unbounded collection of customer data and the exclusive exploitation of this data is a consequence of this imbalanced relationship. It creates an “unfair competitive advantage” for existing companies: companies with big data vaults can offer better products, and thus further optimize their data base.
One of the core goals of a decentralized insurance platform is the disruption of this circle, giving back to customers the ownership of their data.

2.2.5 Summary

The three elements described above; pooling or risk, risk transfer, and efficient administration are necessary. You can’t have insurance without each of them.
For the purposes of this paper, I will call them:

1. expected value of the risk
2. capital costs for long tail risks
3. transaction costs

As we have seen, a community may not wish to generate profit from the first element. The second element yields a risk fee for binding capital which depends on the structure of the particular risk: It is typically lower if the risks are granular and uncorrelated; it is typically higher if the risks are clustered or correlated. The third one depends on the complexity of the process. A simple and highly standardized insurance “product” has a smaller transaction complexity than a more complicated, non-standardized product. This will reflect in lower transaction costs.

---

more people are needed in the center to organize the people at the rim (the “management”). Furthermore, to limit internal inefficiencies, companies need a plethora of control mechanisms (that’s the old style) or complicated incentive systems (that’s the more modern way).

[9] https://assets.kpmg.com/content/dam/kpmg/au/pdf/2016/general-insurance-industry-review-2016.pdf

The three elements are completely independent of the underlying technology, economic environment or currencies. They are the atomic building blocks of every risk-sharing system[10].
As an additional aspect we have seen the information asymmetry which is inherent in the traditional insurance systems, and which is undesirable.

The distribution of expected value (element 1) and capital costs for long-tail-risks among participants (element 2) is inevitable and not specific for a blockchain solution. Therefore, let's focus on the third element.

Blockchain is essentially - among other aspects - a way to solve the transaction cost problem without firms. Without the "design pattern"[11] of firms, transaction costs are subject to combinatorial explosion. The coordination costs for n participants are roughly of Order O(n2) and firms reduce this to O(n). Because of this huge gain in efficiency, firms have many ways to hide profits in the transaction costs, and on the other side internal inefficiencies don't show up fast.

Transaction costs also appear in another context: regulations, which are deemed necessary to protect customers in a context with built in conflicts of interest. Regulations form a very effective "competitor" barrier to entry. While insurance companies often complain about the burdens of regulations, they actually don't have much interest in reducing these burdens, as they discourage new competitors from entering the market.

2.3 Blockchain helps to solve issues of traditional insurance

While the current insurance business has evolved over centuries, and is optimized in many aspects, we have seen that it has severe shortcomings to the disadvantage of customers. We will outline some properties of an alternative system, which remedies these shortcomings.
First, an alternative system should of course offer the basic ingredients of any insurance system: covering expected losses, covering long tail risks, and covering of necessary transaction costs. Obviously, we need ways to capitalize such a system, and we need a system to reduce transaction costs to a minimum. Transaction costs

[10] There is a fourth element - reinsurance. The purpose of reinsurance is to reduce the cost of risk diversification by categorizing and securitizing different risks. Reinsurance and "wholesale" risk transfer enabled by reinsurance adds another layer of complexity, and therefore we won't discuss reinsurance in this paper.
[11] On the importance of design patterns, see also: Design Patterns, by the "Gang of Four" https://www.pearson.com/us/higher-education/program/Gamma-Design-Patterns-Elements-of-Reusable-Object-Oriented-Software/PGM14333.html#

cannot be eliminated completely. But open markets have proven to be a solution for these challenges, and therefore, we propose a market-based approach with two components:

- an open marketplace for capitalization of risks
- an open marketplace for insurance related services

This is where blockchain comes into play: a decentralized solution on blockchain implements such open marketplaces in a way that is collusion resistant and has no single points of failure. We can watch the emergence of many such marketplaces for different domains, like computation, file storage, exchange of assets; and insurance is just another domain in this respect.

More specific, blockchain helps to solve four main problems which pile up costs in traditional insurance companies:

1. Coordination ("managerial") costs.
2. Conflict of interest between customers and company.
3. Information asymmetry between customers and company.
4. Access to risk pools
5. Composability (money Legos)
6. Access to the crypto based capital market

Advantage 1. In traditional firms, you have two types of employees: the first group is doing the actual work, the second group is coordinating the whole system. The larger a company grows, the more energy flows in the second group (like a circle, the first group forms the rim of the circle, the second the area; the larger the circle, the less efficient are the processes, and the more energy flows into the coordination of the coordinators). Blockchain helps reduce these coordination costs. Instead of a posse of managers, "smart contracts"[12] acts as trustless hubs between the agents at the rim of the system, and thus eliminating most of the costs and the inefficiency of the management.

Advantage 2. In a traditional insurance company, the company "owns" the whole process, including the tasks which tend to raise conflicts of interest between customer and company. A perfect example is claims management: The claims manager has the explicit goal of minimizing payouts for damages, because he is employee of the insurance provider! Of course there is a guild of "independent" appraisers and experts, but who pays them?

---

[12] Some blockchains like Ethereum (which we use) enables programs (called "smart contracts") that are un-censorable, immutable, and permanent. These smart contracts can interact with each other to perform a wide variety of actions, including financial and escrow transactions. This makes possible direct and transparent interactions between two parties who may be and may remain anonymous, that previously required a third-party intermediary to be effective. The term was originally coined by Nick Szabo, but in a slightly different meaning. Note: The above definition was thankfully supplied by Ron Bernstein, who was not successful in finding the original author - please contact us if you are the author.

Blockchain solves this conflict of interest, by enabling truly independent experts (who for example may be publicly ranked by their reputation for efficiency or fairness), and whose work is independent of the insurance provider, as well as transparent and auditable by the whole community.

The same is valid for another area, where the conflict of interest is (intentionally) not obvious; consider Product Design. An insurance company has a big advantage over customers, because they can design products in a way which perhaps unfairly maximizes revenues (sales) and minimizes payouts (expenses).

For example if a customer expects a payout from an insurance policy they bought for a particular “event” but the insurance company does not provide the payout because the company maintains that the policy bought doesn't actually cover that “event”, the customer experience is severely degraded and trust is eroded between consumers and insurance providers.

Advantage 3. Insurance companies collect data and information in huge private silos in proprietary ways, and the data is often not shared. This information asymmetry is a source of inefficiency and the origin of high transaction costs.
The experience of companies in analyzing this data is considered one of the key differentiators in the market. Decisions based on this data are not transparent and difficult to challenge due to the lack of insight into the evaluations.
In a blockchain environment, however, all fundamental data and the decisions based on this data are transparent and objectively validated.

Advantage 4. The risk pools of traditional insurance companies are attractive investment instruments. However, they are not publicly accessible and the profits generated benefit only a small group of investors. Blockchain democratizes access to risk pools by tokenizing risks with "risk pool tokens."

Advantage 5. Composability is the general ability of components of a system to be recombined into larger structures and for the output of one to be the input of another. In simple terms, the best example is Lego, where every piece can connect to every other piece. Within Crypto, composability is the ability of decentralized applications (dApps) and DAOs to effectively clone and integrate one another (syntactic composability), and for software components such as tokens and messages to be interoperable between them (morphological composability).

Advantage 6. Blockchain enables more efficient collateral management through the digitization of the collateral holdings into a single, optimized registry. Additionally, smart contracts can enable the precision of collateral management by automatically issuing margin calls and invoking predetermined rules for each bilateral or intermediary relationship. The creation and digitization of collateral tokens or assets facilitate new markets and possibilities. For instance, digitally represented collaterals on blockchain can be used to redeploy and settle in real-time, eliminating delays between valuation and call.

2.4 Requirements and consequences of a decentralized implementation (??)

2.4.1 Why is insurance a candidate for decentralization?

As a multi-trillion dollar industry dominated by huge corporations, insurance is often confronted with obstacles such as strict regulations, and misalignments of company and consumer incentives, which led to the insurance world often times being inefficient and expensive. The ultimative goal is to avoid cases like customers having to fight for reimbursement from companies whose profits often depend on avoiding paying out in a targeted manner.

Etherisc is building a platform for decentralized insurance applications. The platform can be used by corporates, large and small, not-for-profit groups and insurtech startups to provide better products and services. We aim to use blockchain technology to help make the purchase and sale of insurance more efficient, enable lower operational costs, provide greater transparency into the industry and democratize access to reinsurance.

Blockchain can provide the means to disintermediate the market with a peer-to-peer risk platform that helps insurance return to its roots as society's safety net. We even envisage new groups building their own bespoke insurance risk pools and services on the platform. Etherisc framework enables fully-compliant and licensed insurance products for the emerging blockchain economy.

To offer an alternative to traditional company-centric insurance systems, we can identify some requirements and consequences for implementing a decentralized insurance protocol.

2.4.2 General requirements for decentralized insurance

1. We offer a protocol and not just a (decentralized) application. The range of insurance is huge and far too complex to be covered by a single application.

Some tools are needed to incentivize participants to use it. Promoting "network effects"[13] is one tool that can lead to a sustainable and growing user base.
A policy may cover a particular product, but a single policy will not generate the network effects to create multiple large pools of similar risks necessary to take advantage of the "law of large numbers."
Decentralized insurance only works if the value creation process is disaggregated, there is also a way to divide the disaggregated process among different participants, and the participants can work together concertedly. Our generic protocol defines this way. The (architecture of the) protocol is the only "central" part of the model.

2. A decentralized insurance protocol can partially or fully replace “the insurance companies”, by implementing a standardized set of rules for how stakeholders in the system interact with smart contracts and with each other using the protocol. By this, most of the coordination costs are replaced by autonomous and automated contracts and procedures and enforce efficiency by open market mechanisms. At the same time, a protocol does not impose a fixed set of code to the participants, but allows for flexible extension and interpretation of the basic rules.

3. The development and operation of a protocol needs funding. Even if we can drastically reduce the coordination costs, there are still the costs for the initiation of the system - e.g. acquisition of licenses, development of smart contracts, audits, as well as costs for agents at the “rim” of the system which we cannot eliminate completely. Therefore we need a way to collect these costs from the ultimate customers and distribute them amongst these agents.

4. We also need a way to calculate and distribute the expected value of the risk and the capital costs for covering long tail risks amongst the customers.

Obstacles like these are commonly solved by issuing a token. In this specific case, the DIP protocol token gets in the picture. It is an integral part of a decentralized insurance platform. The role of the protocol tokens is to bind participants to the platform and to assure the quality of the provided services. This will be elaborated further in chapter No. 4.

[13] Network effect is described as the effect that one user of a good or service has on the value of that product to other people. The classical example is telephone: the more people use it, the more valuable the telephone is for all.

2.4.3 Requirements for token

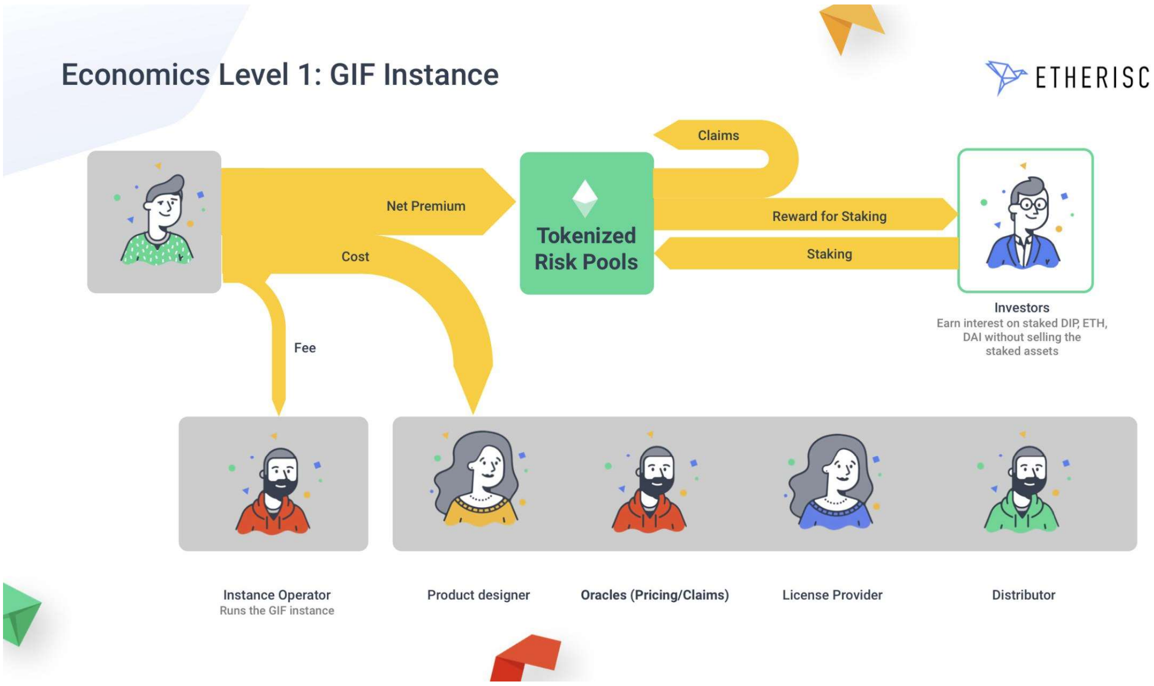
1. **Tokenization**
   is the solution for these requirements - but only if the token is intrinsically required for the protocol to operate efficiently; i.e. “baked into” the protocol itself and usage of the protocol is only possible via tokens. If the token were not intrinsic to the use of the platform, then some new actor could replicate the protocol except without the token, and migrate users to the new protocol without the friction of a purely "rent seeking" token.

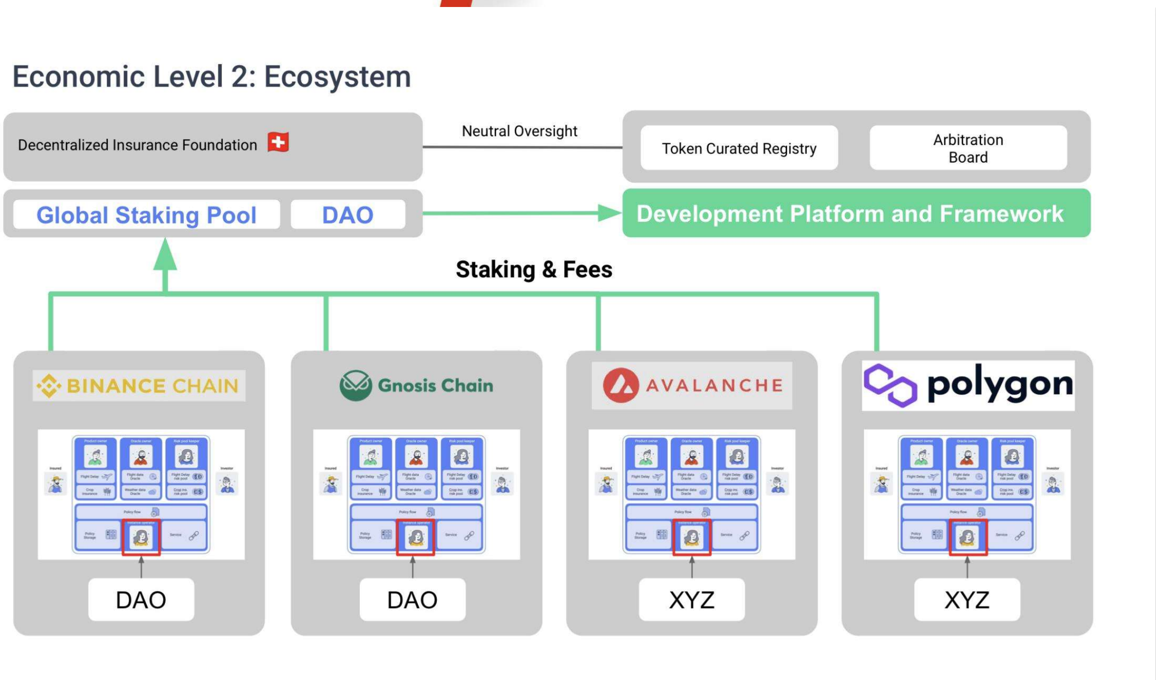
2. **Protocol tokens:**
   For the distribution of the transaction costs we have a different type of token. This token is constructed in a way that incentivizes the use and the efficiency of the protocol: the revenue associated with this token or its price should increase with the efficiency and use of the underlying processes.
   In Chapter 4 , we describe a proposal for a token with these properties.

3. **Risk Pool tokens:**
   For the distribution of the expected value, and for the distribution of the capital costs for covering long-tail-risks, we have a type of token which generates a foreseeable profit. The profit solely depends on the underlying risk structure, the number of risks, their correlation, and so on. The value therefore depends only on the knowledge of the risk parameters (which can be incomplete) and mathematics.
   These tokens e.g. yield a fixed revenue or generate an equivalent rise in price for their owners (which is equivalent). This type of token will be implemented in a second step.

4. Now that we have elaborated the necessary token types, we can backtest if these tokens are “necessary”
   Etherisc built an economic space for decentralized insurance.
   The space has a broad set of participants, customers, service providers, risk carriers, etc., the goal is to incentivize these participants to cooperate and behave well, and in line with the interests of the whole space.
   This space is difficult to build. It comes at a cost.

Ecosystem: Audited and configurable product templates will enable onboarding of new teams in no time.
Economical: Risk pools which are backes by reinsurance pools and which need not be fully collateralized will enable higher capital efficiency.





What adds value to the space:

Building Block	Consists of	Resistance against forks & copycats
Licenses	Formal approval by authorities	Cannot be copied
Operational Model	Infrastructure to run a business	Cannot be copied
Products	Code (Frontend/backend) infrastructure Certifications/Audits Developers Product managers	Tech can be copied, but products are micro-ecosystems with development roadmap, user base, customer support, core development team, supporters and contributors
Users	Customers Supporters Contributors	Cannot be copied
Network	Formal or informal Relations to other projects,	Cannot be copied. Relations to other projects are based on common vision.

5. **Conclusion:**
Only tech can be copied easily. Most of the value-bearing components of the economical space (the value that participants bring) can't be copied easily
The economic space will offer opportunities to generate profits. These profits should benefit those who have participated in building up the space, and they will expect the platform to protect their participation.
Reason:
If you have two identical platforms, one platform with some kind of protection mechanism for its creators and contributors and one without such protection. The platform with protection will, of course, attract more contributors. It will have stronger network effects.
A platform without protection is subject to the "Tragedy of the Commons"

In the prospect of decentralized exchanges, the use of a token is no longer a barrier.

2.5 Protocol

2.5.1 Owner of the protocol, governance

As an open standard, the protocol is a common good, it can be used and implemented by whoever likes it. We will take care that the entry barriers are as low as possible. However, for some portions of the protocol, a certification will be necessary, to reflect regulatory obligations and restrictions. We have founded a swiss based foundation as a legal body, which formally holds the IP rights of the protocol and ensures that the protocol can be used freely. We established a continuous, community-driven protocol improvement process similar to the EIP process for the Ethereum Platform.
The Etherisc Governance Model (EGM), its abstract and core values as well as other subtopics will be further elaborated in Chapter No.5.

2.5.2 Outline of workflow elements of the protocol

- Application for policy
  Process of offering a product and applying
- Underwriting
  Process of accepting a policy
- Collection of premiums
  Payment process, one-time and regular payments
- Submitting of claims
  Process of submitting a claim, via oracle or manually
- Claims assessment
  Process of assessing a claim, via oracle or manually. A claims verification process allows the system to determine which policies are legitimately claimed and to propagate agreed payments to claimants. In the case of parametric insurance, this process references data feeds about insurable events and is (fully) automated.
- Identity Management & Privacy
  Process of KYC and AML, respecting privacy. This may involve private chains or off-chain storage of data.
- Admission / Certification
  Admission of participants to offer products and perform parts of the protocol
- Asset Management
  As funds flow in, we have to responsibly use funds which are not immediately needed.

2.6 Community of customers, users and companies

The success of the platform will depend on a vivid community of users and companies. The token model reflects and supports this community. This community plays a central role in the realignment of incentives. Via tokens, customers can “own” their insurance. The community model facilitates the development of future mutuals and P2P-Insurance models. A community cannot be built from the outside, it has to grow from the inside. However, experience shows that there are some success criteria for communities. Famous open source pioneer Pieter Hintjens, http://hintjens.com/blog:10 has drafted some which we consider to be helpful for an in-depth discussion:

- Quality of mission. A community can only grow by pursuing a worthwhile goal. The goal must be super-individual.
- Freedom of access. The community should not have barriers or walls, it should welcome those of goodwill and encourage participation.
- Well-written rules. If rules are necessary, they should be carefully written and obvious.
- Strong neutral authority. To resolve conflicts, a strong but neutral authority should be in place, which can be incorporated by a governance mechanism.
- Proportional ownership. "You own what you make"
- Infinite spaces. A single large project with many owners does not scale as well as a collection of many small projects, each with one or two owners. Communities grow best when people layer project upon project without limit.
- Measurements of success. In the community, your voice is as loud as the number of people using the project you “own.”
- Tools and processes. Much better tools means a faster, more efficient community.
- Freedom to organize. Let the community participants identify the problems, allocate the resources, and monitor success, precisely without top-down management.
- Transparency. Secrecy enables incompetence, and transparency promotes competence. The more public the organization’s work, the better.
- Unstructures. “Everyone owns what they make” and be prepared to move to a new home as and when needed.
- Scalable participation. You want no barriers at any point, but it must get harder and harder. This makes the community feel like a massive multiplayer game, where there’s always someone better than you, and you just have to try to catch up.
- Full remixability. Every work needs to be remixable.

3 GIF

The GIF consists of building blocks that include the complete value chain: the insured, the insurer, the investor and the instance operator.
First of all, you need insurance products that you can sell. The insurance products have a product owner who designs the products. The insurance products themselves are from smart contracts.
Oracles are an essential part of the GIF for implementing parametric insurance. Oracles provide the necessary data, for example, flight or weather data, to the contracts in a GIF instance.
The risk pool is also a smart contract that keeps track of all details of the risk capital, the amounts paid in as policies and all the amounts paid out.
A GIF instance connects these individual roles and represents a complete executable entity defined by the GIF. Each instance consists of a blockchain's operational set of GIF smart contracts. Different blockchains may run different instances.

3.1 Etherisc-basics about insurance

In the chapter '2.2 Principles of insurance', we used a practical example to illustrate how insurance is created and functions with the insured's participation. In these chapters, we will explain and define insurance from the perspective of Etherisc.

3.1.1 What is insurance?

Insurance is a means of protection against financial loss. It is a form of risk management whose primary purpose is to protect against the risk of possible or uncertain loss. The loss associated with the risk may or may not be financial, but it must be reducible to financial terms.
An insurance company[14] underwrites the risks of the insured. The insurance company can outsource all services, such as sales or data management, to other service providers. The only exception is the actual assumption of risk. This risk must always remain with the insurance company. Therefore, the company and their customers always need to have a proper accounting on which risks they cover and how it is collateralized. Via smart contracts, this can be done in a transparent and auditable way.

3.1.2 What is an insurance policy?

An insurance policy is a contract provided to the insured by the insurance company that sets out the conditions and circumstances under which the insurance company

[14] We use the term “company” here for easier reading. Of course, in DeFi/blockchain applications, a “company” can also be a DAO or a simple blockchain address (EOA)!

will make payouts to cover losses incurred by the insured due to recognized claims. In TradFi, this is typically a legal contract. In our context, a policy is simply a dataset stored on blockchain and manipulated via defined rules by a smart contract.

Let’s look at the lifecycle of a typical insurance policy. Such a lifecycle usually consists of the following chronologically listed sub-steps.

- The customer inquires about an insurance policy. They want to protect themselves against a specific risk by taking out an insurance policy.

- The insurance company examines the customer’s application.
- The application is accepted or rejected.
- In case of rejection, the customer is informed and no further activities occur.
- In case of acceptance, the contract comes to the “underwriter.” The acceptance of the application is called “underwriting.”
- The insurance company commits itself by the “underwriting” to take over the customer's risk and transfer it to itself. It further undertakes to cover the loss if the insured event occurs.
- The customer, for his part, undertakes to pay the premium.
- Both declarations of obligation are documented in a contract. This contract is called the insurance policy.
- If a claim occurs, the customer reports it to the insurance company.
- The claim is checked by the insurance company and accepted or rejected.
- In case of acceptance, the agreed insurance sum is paid out.

It is easy to see that the classic insurance business generates considerable bureaucracy and that many individual sub-steps require manual activities. For example, when a customer files a claim, the insurance company has to check the claim's details manually. This involves work and, therefore, costs.

3.1.2 What is parametric insurance?

Parametric insurance is an agreement between the insurance company and the insured that covers the occurrence of predefined events rather than manually reviewing and compensating for actual losses incurred.

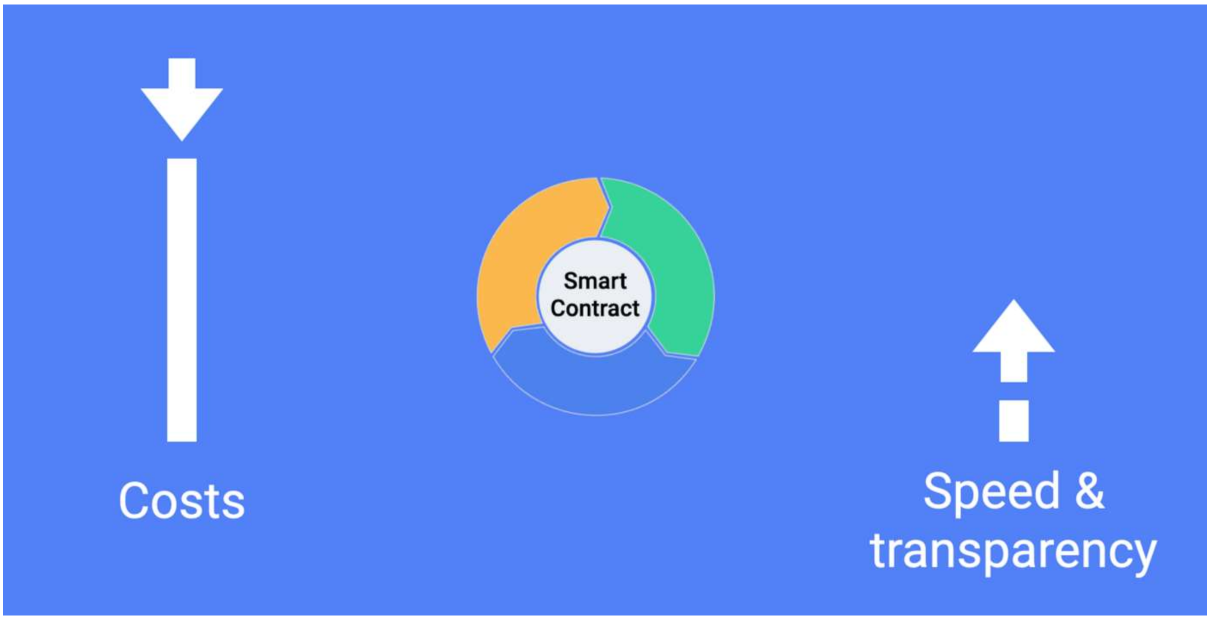
Parametric insurance policies correspond to agreements between the insurance and the insured where the insurance approves payouts to the insured when predefined triggering events occur.

In parametric insurance, loss events (the risks) are defined as functions of underlying indices or parameters that meet the criteria defined by the insurance product. Example indices/parameters include rainfall amounts and wind speeds for insurance linked to weather conditions. In the case of flight delay insurance, the parameter/index can directly be derived from the difference between the actual arrival time and the scheduled arrival time of an insured flight. To make parametric insurance feasible and attractive to all involved parties, the underlying indices/parameters must be transparent, reliable and trusted.

Once such events occur, the insurance directly calculates and triggers a payout to the insured without an often costly claims acceptance process.

The big win of parametric insurance is its potential for efficiency and automation. Claims handling, one of the most complex and costly parts of the insurance business, can be reduced to a simple and fully automated process.

3.1.3 What are the advantages of blockchain in insurance?



There are several benefits that blockchain technology can provide to the insurance domain. Some of these benefits are directly related to the foundations of blockchain technology.

- Transparency and accountability for record keeping. Information regarding policies, claims and payouts may be stored on-chain. Once on the chain, they can neither be deleted nor changed without proper permission, and each time data is updated or adjusted, the original data is kept in the history. An entire audit trail is available and transparent for all data.

- Minimize friction and transaction costs for payment handling.
- Create new markets/opportunities by opening risk pools. The transparent pooling of large numbers of insurance policies of a particular type provides the opportunity to open up this market to a wider audience.
- These new markets also include the option to trade risks in small quantities, so called “Risk Pool Tokens”.

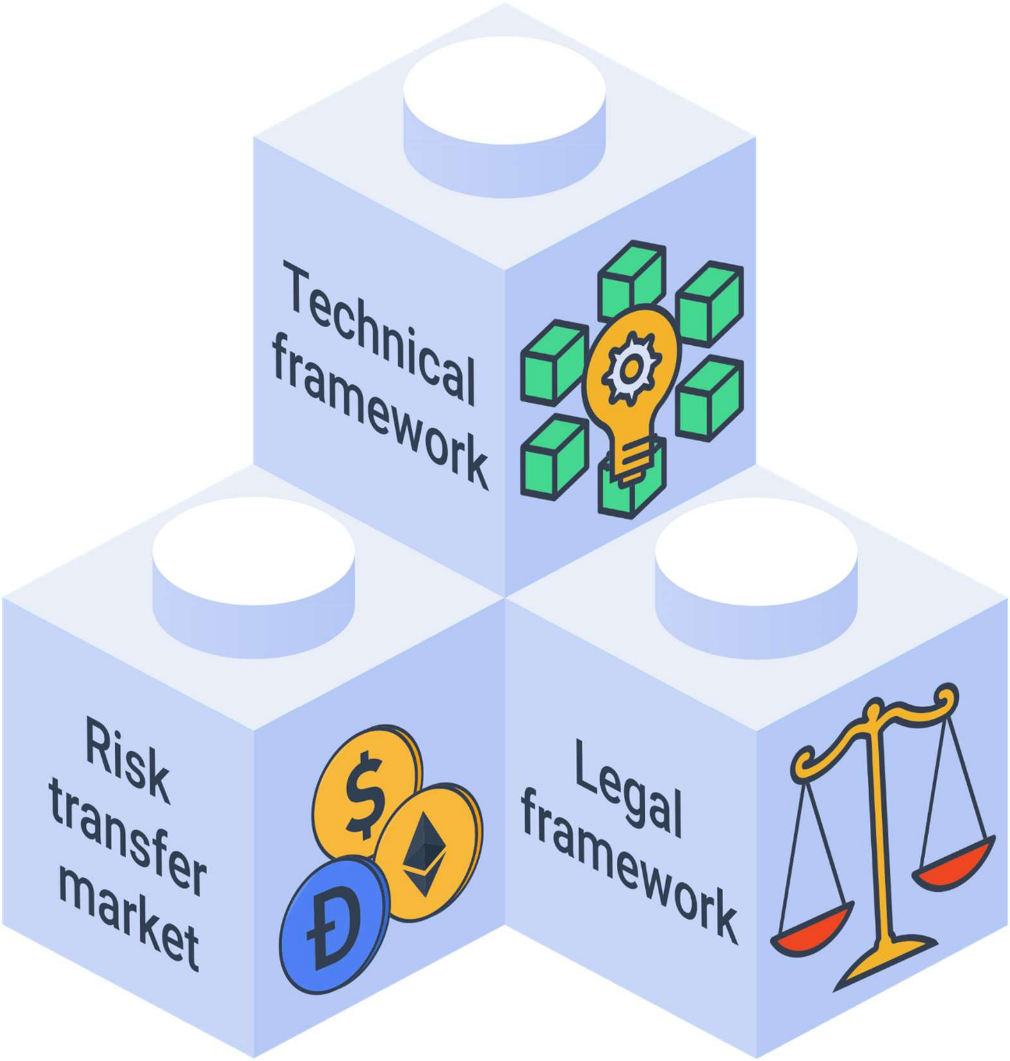
Blockchain technology can provide a lot of value, especially for parametric insurance.

- Providing this central data in a trusted way to the blockchain world will be managed through oracle services, making it very hard/too costly to inject manipulated index/parameter information into smart contracts implementing parametric insurance policies.

- Once the index/parameter feed is provided to policy contracts, parametric insurance will fully automate claims and payout handling.
- Immediate payouts. Running in a blockchain context and having automated claims/payout handling allows for near-real-time payouts.

3.2 The Etherisc model

3.2.1 The three pillars of the Etherisc ecosystem



Risk transfer market

Raising capital to back the technical guarantees is done by investors. Investors will lock a certain amount of DIP tokens, also known as “staking.” The staked DIP tokens are a prerequisite to be then able to invest the actual risk capital in DIP or stablecoins.

The community of DIP token holders created the entire Etherisc ecosystem. Therefore, we will demand that parties who profit from the ecosystem own a share by holding and staking DIP tokens. This idea is borrowed from the space of cooperative enterprises. It reflects that the Etherisc ecosystem is a public good that needs to be protected from the “tragedy of the commons.”

Legal framework

Insurance companies are highly regulated worldwide for good reasons, to protect customers and investors. A great deal of legislation has been enacted for this purpose in most countries. Concerning jurisdiction, a general distinction can be made between the American, European and Anglo-Saxon regions.

But even within these regions, each country has different legal and monetary frameworks. Etherisc engages with local regulators to help create an efficient regulatory environment for blockchain based insurance. Etherisc supports interested parties and helps to guide the coordination process with the relevant agencies and ministries.

The financial and organizational hurdles to establishing a new insurance company are high. For countries like Germany, Etherisc offers a new legal model where the legal claim is exchanged for a technical guarantee using blockchain and smart contracts. Thus, the provider — in this case Etherisc — is no longer subject to an insurance company's legal and financial requirements. Still, for each project, product and jurisdiction, the legal framework has to be considered and the product owner is responsible for the proper implementation. The Etherisc team has accumulated a lot of experience in this field and is happy to share these insights with platform users.

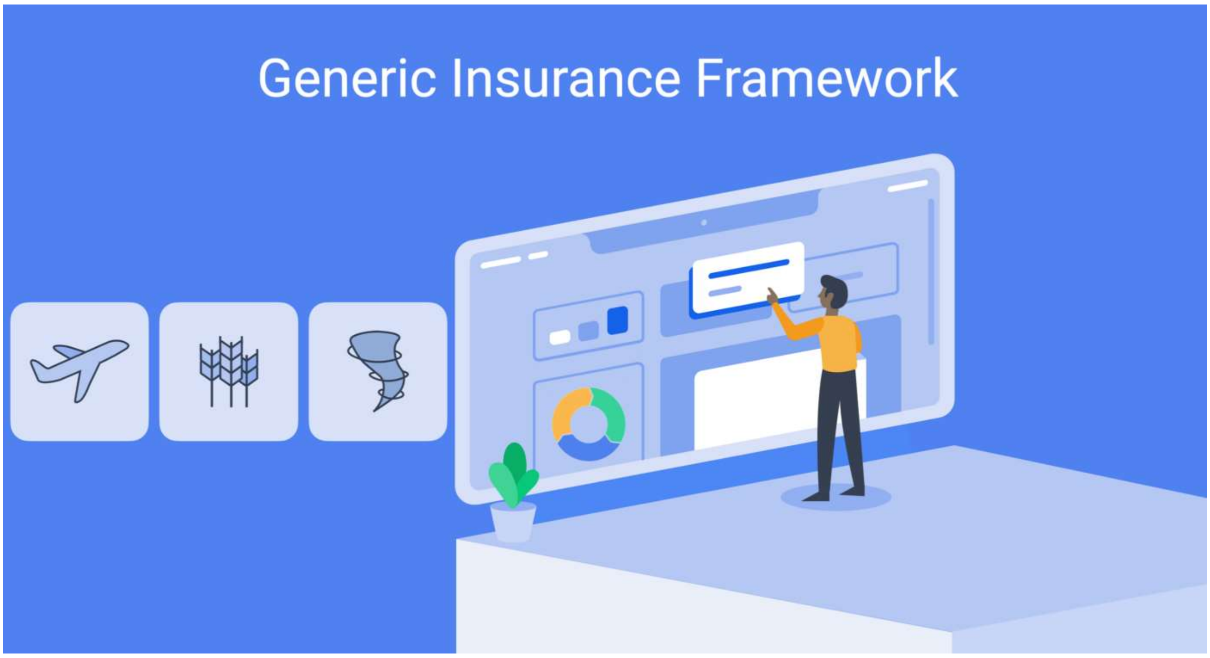
Technical framework

With the Generic Insurance Framework (GIF), it is technically possible to model insurance policies individually.

Developed and maintained by Etherisc, the GIF allows to model, deploy and operate insurance products based on blockchain in a decentralized and transparent way.

Using the GIF, interested parties can quickly implement and securely operate their insurance products.

3.3 What is GIF?



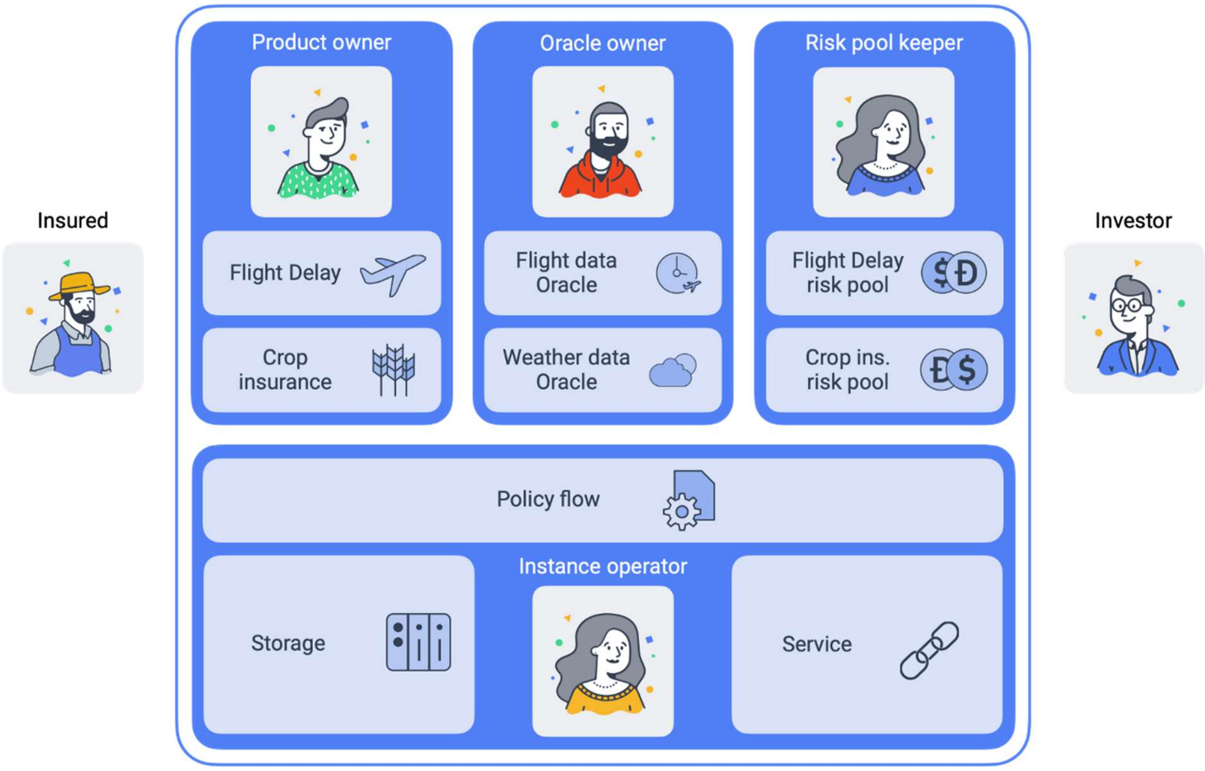
GIF is an acronym and means generic insurance framework. At its core, it consists of a collection of open-source smart contracts that implement generic functions of the lifecycle of insurance products and policies. Thus, GIF enables the modeling of a wide variety of insurance types.

Processing steps that run similarly in all products have been identified and made available as modules to design insurance products quickly and easily. Thus, only product-specific aspects, such as pricing, etc., need to be implemented for each product.

3.3.1 GIF and GIF instances

To operate insurance products, including selling policies, collecting premiums, calculating trigger events and handling payouts, a complete execution environment is needed in addition to the smart contract collections that define products and policies.

This execution environment — called a GIF instance — may be seen as a comprehensive platform or marketplace in which GIF-based insurance products are managed and operated. Our goal is for a GIF instance to be used by many different and independent organizations providing various insurance products. The figure below provides an overview of the stakeholder roles involved with a GIF instance.



3.3.2 Participants on the platform

Insured/Customer

The insured / customer is the policyholder who wants to transfer his risk to the risk pools. Third parties can offer payment gateways and integrations which remove the necessity to own cryptocurrency from the end customer.

Investor

Investors are interested in participating in risk pools to balance/diversify their risk portfolios. Investors provide collateral for risk pools in return for interest payments.

Oracle owner

One of the most promising applications of a decentralized insurance space is the way data is collected and managed. The oracle owner provides oracles that interface between the blockchain smart contracts and external data sources. In the case of flight delay insurance, the oracle informs the smart contract whether the flight landed in time, how much it was delayed or if it was completely canceled.

Product owner

The product owner designs and operates one or more products. This would be an insurance company or an MGA (managing general agent) in the traditional insurance industry. Due to the multi-client capability, a product owner can use all oracles located on the respective platform by the oracle owners.

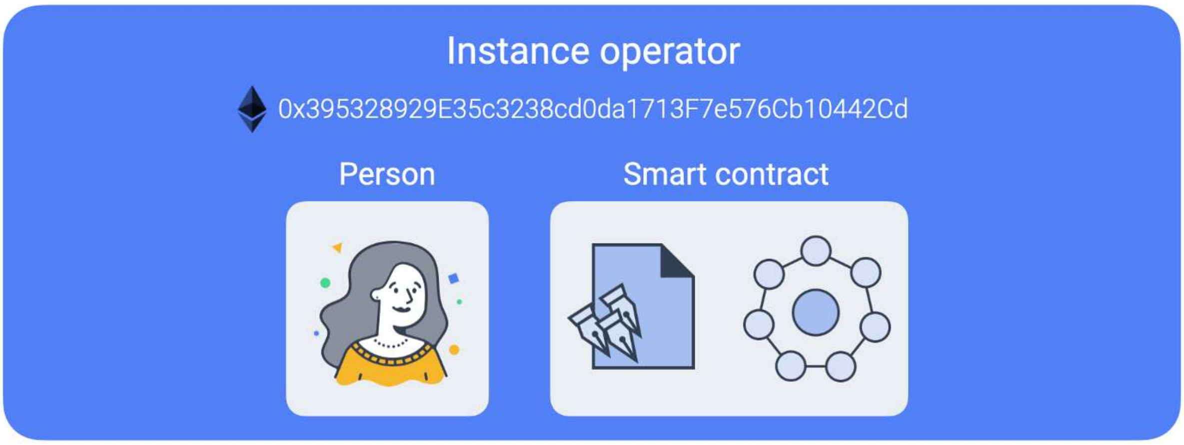
Risk pool keeper

A risk pool keeper manages one or more risk pools. A risk pool is a smart contract that allocates ("pools") several risks, represented by policy objects, to risk capital.
Risk pools collect collateral that risk investors invest. Losses are paid from the risk pool. Therefore, the capital in the pool is at (default) risk. Investors can top up their investments and also withdraw their funds.

Instance operator

Any complete deployment of a GIF framework is called a "GIF instance."
There will always be at least one complete instance of the GIF operated by the Etherisc project, but in principle, anybody can deploy a new GIF instance. The instance operator is the crucial role that operates a specific GIF instance.

The primary tasks of the instance operator are the administration of products and oracles and a few other basic actions. Any GIF instance is multi-client capable, which means that any number of product owners and oracle providers can be operated and administered on one GIF instance.



The instance operator is represented by an Ethereum address. The instance operator can be a natural person owning the private key of that address or a smart contract — either a multisig or a DAO structure. This enables an entirely decentralized operation of any GIF instance. One address can, of course, manage several independent GIF instances.

It is the declared goal of the Etherisc Project that GIF instances are controlled in a decentralized way - either by multisigs or by DAOs with their own governance structure - and that they are controlled by the platform's stakeholders (customers, product owners, oracle owners and risk pool keepers). In Chapter 5 we discuss how the ecosystem can incentivize this development.

3.4 Generic lifecycle functions in GIF

3.4.1 Concept of components

Each GIF instance manages different components. A component is a specific smart contract witha certain core functionality. The components can represent different core objects.
The core objects are:

- Products

- Oracles
- Risk pools

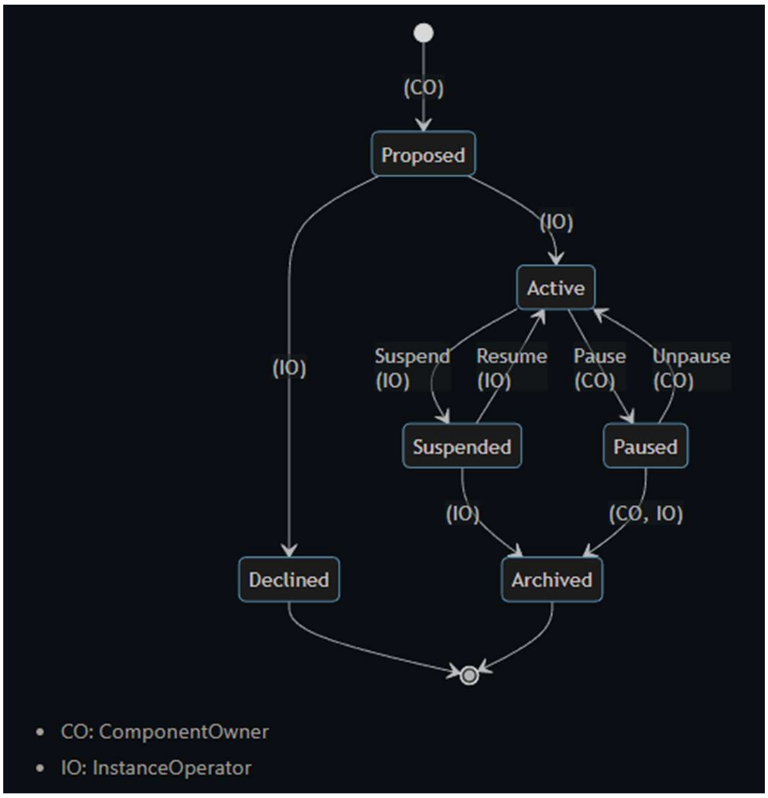
All components and thus the objects they contain can assume identical states and have the same life cycle but can differ significantly in terms of lifespan.

3.4.2 Component roles and lifecycle

Two roles can determine the life cycle of a component.

Component owner
A Component owner can be an oracle owner, a product owner, or a risk pool keeper, depending on which core object he manages.

Instance operator
The instance operator runs one or more GIF instances.



A component in the GIF is always in one of the following states:

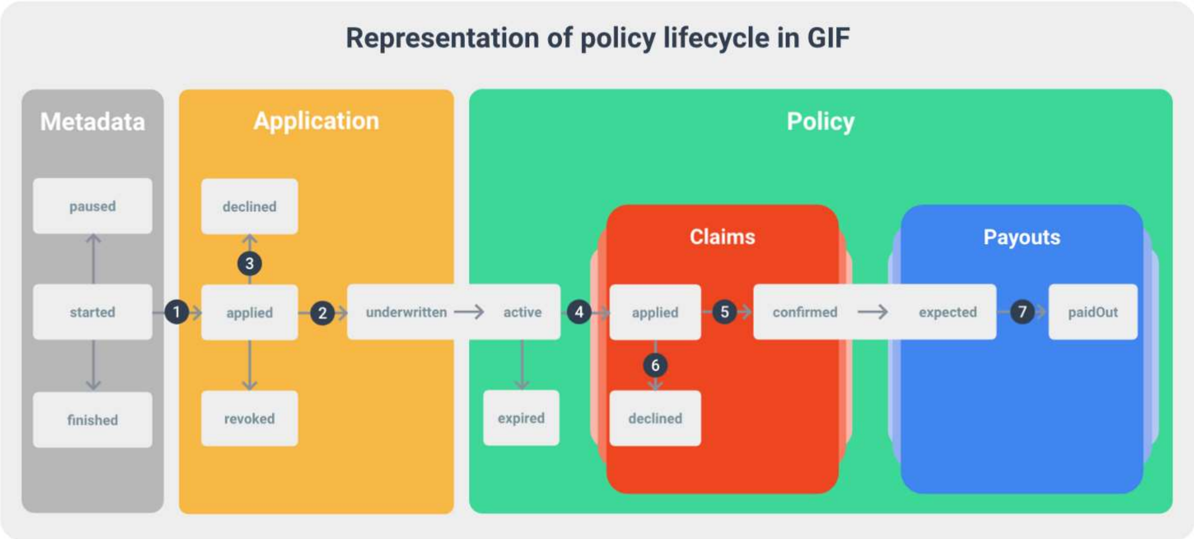
- Created
- Proposed
- Declined
- Active
- Paused
- Suspended
- Archived

The transition between these states and the roles which those can be triggered by are described in the above diagram. The lifecycle of a component starts with its development and deployment on the blockchain. The component owner can implement their specific requirements in the component's smart contract or use the generic functionality of the GIF components. In the next step, the component is registered, approved and activated by the instance operator in the GIF instance. The instance operator can also decline a component. The component is then deleted.
In the event of approval, the instance operator continues to check the technical and procedural details. The instance operator can also outsource the verification to an independent audit.
Another condition is that the component owner must contribute a certain amount of DIP tokens to be allowed to operate in the GIF instance.
If the component is active, it can be used until it is set to either suspended or paused. The difference between suspended and paused is that only the instance operator can suspend a component or resume it from suspended to active. The component owner

can set a component to paused, the component owner and the instance operator can unpause the component. If the component is inactivated (pause, suspended) and not reactivated (resume, unpause), it is not deleted but archived.

For each type of component (products, oracles, risk pools) we provide sample implementations which can be used as a starting point.

3.4.5 Policy lifecycle



Independent of the specific product, each policy processed on the GIF instance has a lifecycle. Typically, a policy undergoes several state changes during the lifecycle. While any product designer could implement his own lifecycle (in our terminology, the life cycle is called “PolicyFlow”), the GIF offers a default lifecycle which should be sufficient for most use cases. This generic life cycle is called “PolicyFlowDefault.”

The “PolicyFlowDefault” lifecycle offers the following functions:

- newApplication (to generate and store a new application from a customer)

- underwrite (to sign an application and create a new policy)
- decline (to reject an application)
- newClaim (to generate and store a new claim in case of loss)
- confirmClaim (to confirm a claim and create a payout)
- declineClaim (to reject a claim)
- payout (to confirm and initiate a payout)

3.4.4 Payments

The GIF instance is agnostic to the way payments are made. Pure crypto payments can be made directly to the product contract, while fiat payments need a fiat gateway and an external banking or credit card infrastructure. The core team can request information on how to implement fiat gateways.

3.5 Introducing the GIF monitor

The GIF system is entirely transparent for blockchain experts, but it can be difficult for non-blockchain experts to understand. That's why we developed the GIF monitor to give everyone an overview of what's happening on the blockchain of a GIF instance.

3.5.1 What is the GIF monitor?

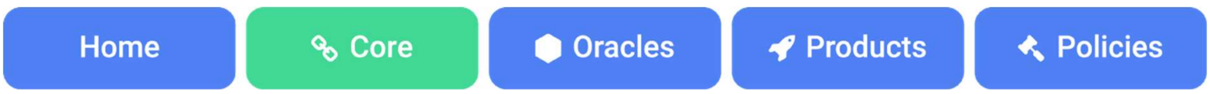
The GIF monitor provides a structured overview of all generic building blocks available in the GIF framework for creating and operating an insurance product. You can view all events and business transactions of the complete instance.
The GIF monitor provides all this information transparently and in real-time online. The information is read from the blockchain and the GIF framework.

3.5.2 Menu items

The URL https://gif-monitor.etherisc.com/ takes you to the 'Home' area of the GIF monitor. In the menu bar, you can choose from the following menu items.

In the 'Home' area you can click directly on the menu items in the menu bar and then select the corresponding menu item in the drop-down menu.

Core



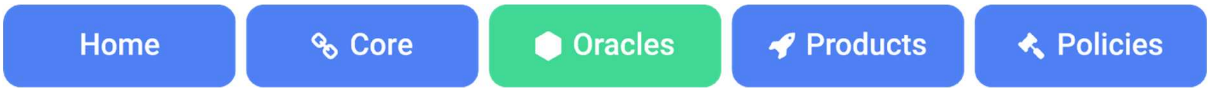
The 'Core' area is by far the most extensive area. It displays the available GIF instances, the GIF core contracts per instance and the events of these core contracts. The core area shows the complete core contracts that each user can use.

Here you find the blockchains the instances use, such as xDai or Ethereum.
By clicking on an instance, you will get detailed information like the instance ID, name, name of the blockchain, chain ID and status (active or not). Each instance is identified by its registry address. GIF is multi-chain capable and can run on all significant Ethereum-similar blockchains.

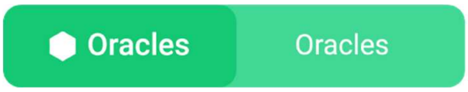
In this section you will find all 14 GIF core (smart) contracts. Each core contract provides essential functionality to a GIF Instance.
You can click on the contract name for all GIF core contracts to get to the contract details. Here you will see on which instance you are, the instance ID, the address on the blockchain, the name of the core contract, and the detailed contract functionality as described by its contract application binary (ABI) interface.

Here the contract events of the GIF core contracts are displayed.
Contract events are emitted by smart contracts during code execution and permanently stored on the chain. Events are primarily used to document significant changes in the data of the smart contracts, for example the change of status.

Oracles

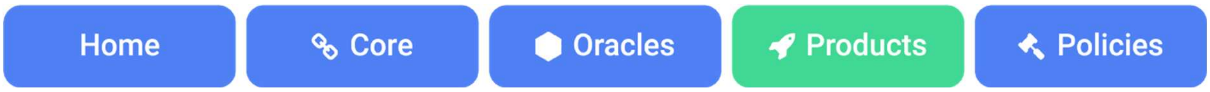


The available oracles are displayed in the 'Oracle' area of the GIF monitor.



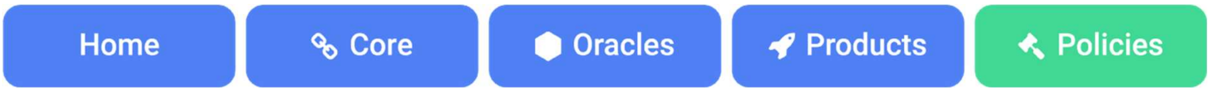
On this page, you will find all oracles available on the platform. Here you can view all input and callback formats as a product owner. In addition, the appropriate oracle can be requested from the Oracle owners.
By clicking on an oracle, the GIF monitor displays the details.
Of course, individual oracles can also be implemented on request.

Products



In the 'Products' area, all products are listed that have been created in the framework.
By clicking on a product, the details are displayed.

Policies



In the ‘policies’ area you can find information on every phase of the life cycle of a policy. Starting with information about the product, metadata, application, policy, claim and payout. Depending on the policy's life cycle, more or fewer information blocks are displayed.

4 Tokenizing Insurance

You can tokenize almost anything. Some things make less sense than others. We are convinced insurance is an increasingly hot topic around tokenization.

4.1 The DIP Token

DIP stands for the Decentralized Insurance Protocol and the Decentralized Insurance Platform. DIP is the native token of Etherisc issued by the 'Decentralized Insurance Foundation' (DIF) based in Zug/Switzerland.

During the Etherisc DIP TGE (Token Generating Event), DIP tokens have been created on the public Ethereum mainnet. We preferred using this term over “ICO” or “token sale.”

Quick facts about the Etherisc DIP TGE

- Hardcap: 30 Million USD
- Total Supply: 1 Billion (10^9)DIP
- Tokens distributed during TGE: 300M DIP (= 30% of total supply)
- Exchange Rate: 1 DIP = 0,10 USD
- Only registered contributors were able to participate in the Etherisc DIP TGE

Participants (not customers) need tokens to join the platform's "ecosystem." Depending on the service offered, a different number of tokens are required or offered to use the platform or offer services on the platform. Simple services require a small number of tokens, while complex or critical services require more tokens. The amount of tokens that must be provided as collateral or rewards depends on the potential damage caused by participant misconduct or violation of the platform terms. In the future, these parameters will be the basis for controlling the platform. Participants have voting rights based on the tokens they own.

The DIP token serves as collateral and a representation of the network's tangible and intangible value ~~of the network~~ similar to the way financial resources serve to secure operating resources in a cooperative.

4.2 The GIF Staking philosophy

When you buy an insurance policy, you expect a payout in the event of a claim. To ensure that there is always enough liquidity to start or continue selling policies and to service all payouts, we will set up a system with two corresponding risk pools and a risk pool that spans all. This ‘umbrella’ risk pool has the function of a reinsurer. The two corresponding risk pools will collect premiums for all policies sold and additional liquidity provided by investors. The ‘umbrella’ risk pool exclusively contains staked tokens. Each well-designed risk pool is subject to an actuarial model for the insured risk and thus determines a certain probability of default for each policy.
The product owner can use statistical methods to calculate the absolute risk for all policies in a given product assigned to a particular risk pool.
Since Etherisc claims to implement decentralized insurance, our risk pool implementation must not depend on the continuous management of a centralized party but be largely or completely trustless. This means that the smart contract code contains -all the risk pool functionality,~~.~~
~~Almost?~~ except ~~T~~the calculation method for profit/loss of the tokens staked~~toked~~, which can be very complex.~~,~~ ~~which is why~~ For this reason, it does not take place on the blockchain, but rather centrally, and is brought to the blockchain via an oracle. This method has the great advantage that the calculation method can be updated if necessary, for example, if the risk pool keeper DAO decides to do so.

4.2.1 Staking for Risk Pools

We define staking in the decentralized insurance context as:

"The process of attracting and binding capital from investors to specific risk pools to cover their long-tail risks."

In its simplest form, a risk pool can be a multisig funded by interested parties, investors or product owners.
Premiums from the purchase of policies are paid into the risk pool, and claims are covered from the funds of the risk pool. In this ~~most~~ simplest form, investors have little control over their investments. Therefore, the investors must trust the risk pool and its keeper. We have developed a concept of creating trustless risk pools.
Investors are rational actors who make decisions based on evidence for this concept. An investor chooses an investment option based on risk tolerance and portfolio structure.
Alternatively, the investor selects his investment according to ethical aspects such as environmental friendliness, climate neutrality, or social commitment. If necessary, he accepts a lower profit to offer insurance to small farmers, for example. We will always encourage the implementation of green & fair products on the GIF!

4.2.2 Trustless Risk Pools

For a trustless risk pool to function, methods must be implemented that technically and transparently guarantee that the interests of both the insured and investors are met.
For the insured, this means that we can prove that the risk pool will always be able to fulfill claims.
For investors, this means they receive a fair share of the profits made and can decide for which risks they will engage with their funds. This results in the need for the system to run entirely on the blockchain.
We have therefore implemented "epochs" into our risk pools. The duration of an epoch depends on the product.
For each epoch, all policies sold are treated the same. This step reduces the complexity massively.

4.2.3 The basic idea of risk pools and rewards in insurance

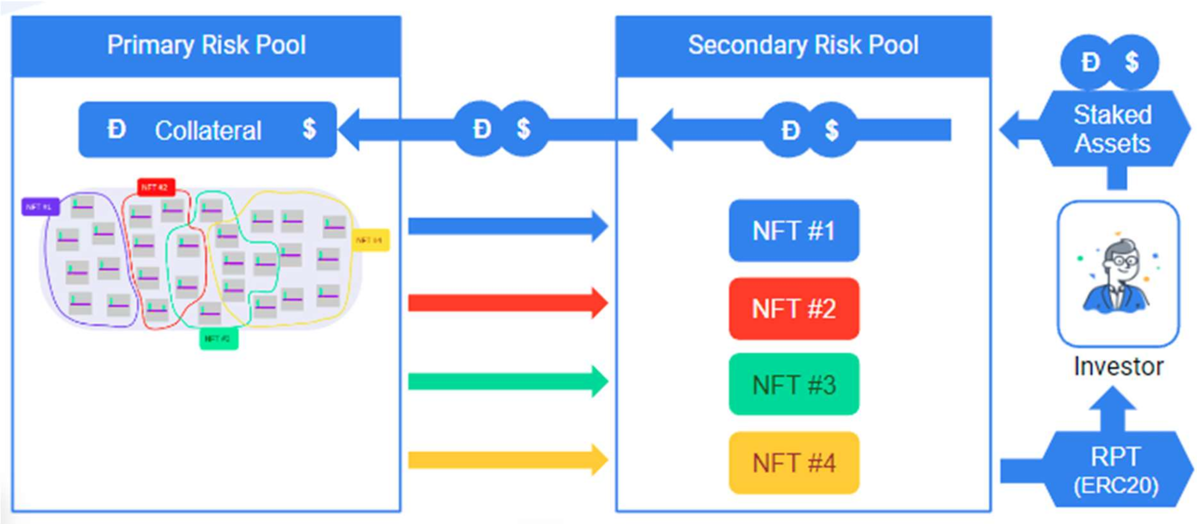
The core economic process of insurance, the transfer of risk between insureds and investors, is implemented in the GIF framework through risk pools.
While the standard risk pool template includes all the core functions for processing premiums, claims, deposits, payouts, and returns, the template leaves maximum flexibility for risk pool operators to design their pool's economic model to appeal to insureds, product owners, and investors.

4.2.4 Core functions of a risk pool

- Receive premiums in the form of native tokens or stable coins.
- Receive investment deposits in FIAT currencies or tokens as specified by the risk pool keeper.
- Manage investment deposits. An investor must have insight into the status of his investment at any time.
- Payout claims in case of loss.
- Processing investment withdrawals. This mechanism is designed so that the risk capital can be paid out only when it no longer serves as a security of concluded policies.
- Process profit distribution. A significant part of the paid premiums is distributed as profit to the investors depending on the investment amount, the period of the deposit and the risk taken.
- Autonomous control of risk pool parameters. The size of risk pools depends on the demand for the underlying product. We will provide mechanisms that allow autonomous control of risk pool parameters.

4.3 Implementation of risk Pools in GIF

The standard risk pools initially consist of a primary risk pool (PRP) and a secondary risk pool (SRP). This combination of two risk pools provides complete flexibility for insurance products and investors.



4.3.1 Primary, secondary and global staking pool

The primary risk pool (PRP) manages the premiums paid by insurers in stable coins (USDC, in this example, xDai at Flight Delay) and the DIP tokens staked by investors. The PRP generates risk pool NFTs in exchange for the DIP tokens or other assets staked by the investor.

Assets such as DIP tokens can also be managed in the secondary risk pool (SRP). Furthermore, the risk pool NFTs are administered and records the created and issued risk pool tokens (RPT).

An investor does not transfer his DIP token directly into the PRP but into the SRP. The deposited DIP tokens are collected and transferred to the PRP at the end of an epoch, and the PRP generates a new risk pool NFT; the owner is the SRP. The investor receives the equivalent value of his deposited DIP tokens in risk pool tokens (RPT) minted by the SRP.

The global staking pool spans all primary risk pools of a GIF instance. Comparable to reinsurance, the pool steps in if, for example, black swan events lead to the insolvency of a primary risk pool, then the parachute pool steps in. Investors can also stake their tokens and stable coins in the parachute pool.

4.3.2 Risk pool token

If the investor stakes assets in the risk pool, he receives risk pool tokens (RPT) in proportion to his share in the risk pool. New risk pool tokens are minted when new capital is deposited into the pool. In general, the staked DIP tokens remain in the particular risk pool. In exchange, the capital provider receives RPT, which are ERC20 tokens and, therefore, tradable. The investor can sell the RPT or exchange them for the DIP tokens at the desired time. Specific RPT are minted for each risk pool.

4.3.3 Risk pool NFT

The NFTs are thus a tranche, meaning all investments from one epoch and include the risks of the insurance policies concluded in the epoch. The NFTs remain in the SRP and the RPTs in the investors' wallets. When all stable coins in the PRP have been paid out to settle claims, the staked DIP tokens in the PRP are exchanged for stable coins and further claims are paid out.

At the end of an epoch, an epoch closing takes place. This only occurs when all policies concluded in the epoch have expired.

4.3.4 Why Epochs?

We want investors to deposit and terminate as quickly and efficiently as possible. But the protection of other investors and policyholders is also essential to us. So we have found a compromise that benefits all parties, the Epochs.
Epochs massively reduce computational complexity, a limiting factor in smart contracts due to blocking gas limits. The epoch concept can execute every transaction with a fixed gas cap.

4.3.5 Single-sided/double-sided staking

In the first release of our risk pool, we will offer single-sided staking only, but here is our current plan: with Etherisc, a base amount is first paid/staked in the DIP token. So only when you have paid the base amount in DIP tokens can you contribute risk capital in the form of risk pool tokens (RPTs). Investors will be able to stake many assets depending on the requirements of the risk pool keeper in the risk pools in the future.

4.3.6 Credit rewards and payment losses

The standard offered by the generic insurance framework is simple. Premiums are added to the risk pool (after deducting costs) and increase the value of the risk pool tokens. Payouts are paid from the pool and decrease the value of the risk pool tokens. In the standard implementation, profits initially remain in the pool. Profits are realized the moment capital is withdrawn from the pool.
Investors receive their premiums in the epoch in which the contract is concluded. The premiums paid by policyholders are credited proportionately in the ratio of personal risk capital / total risk capital. Any refunds in the event of a claim are shared proportionately by all risk capital providers who have contributed to the premiums since the policy's inception.
The cancellation process is carried out in reverse order, taking into account the remaining terms of the risks assumed.

5 Etherisc Governance Model (EGM)

5.1 Abstract

1. The purpose of the Etherisc Governance Model (EGM) is to create an effective self-regulatory mechanism for the Etherisc ecosystem. Etherisc considers a baseline of rules and procedures as necessary to ensure that:
    a. The platform operates in a way that is consistent with the rules and recommendations of the Decentralized Insurance Platform (DIP) protocol.
    b. Participants of the platform conduct business in the interest of the good of the commons, while safeguarding the interests of customers and investors.
    c. Market integrity is preserved, meaning no market abuse and all platform participants have equal access to accurate and transparent information.
2. Consistent with a decentralized infrastructure, regulation should be carried out by the community rather than a sole entity. Additionally, rules need to be enforceable to incentivise compliance. For rules to be enforceable, there needs to be an element of staking.
3. Beyond a smooth-functioning ecosystem being an end in itself, the EGM will be instrumental to strengthening confidence in the Etherisc decentralized insurance platform and support growth and massive adoption.

5.2 Core Values

Any system of rules requires a set of underlying principles and “values”. Both the set and the meaning of these values is necessarily to a certain extent fuzzy and cannot be fully captured by any formal definition. Some people would e.g. emphasize other values not listed here, or put it in different words. However, these rules have been proven to be helpful in other contexts which rely on decentralization and collaboration. They serve as general guidelines to derive more precisely defined rules and requirements.

1. **Respect**
   Each platform user, actor, stakeholder should respect and value differences. We promote inclusiveness and treat others with tact, courtesy and respect. We abstain from and actively discourage discrimination in all forms.
2. **Partnership**
   The Decentralized Insurance Platform is based on strong, voluntary partnerships. The Platform will always encourage partnerships and

cooperation. Each participant should be able to benefit from the evolving partnerships.

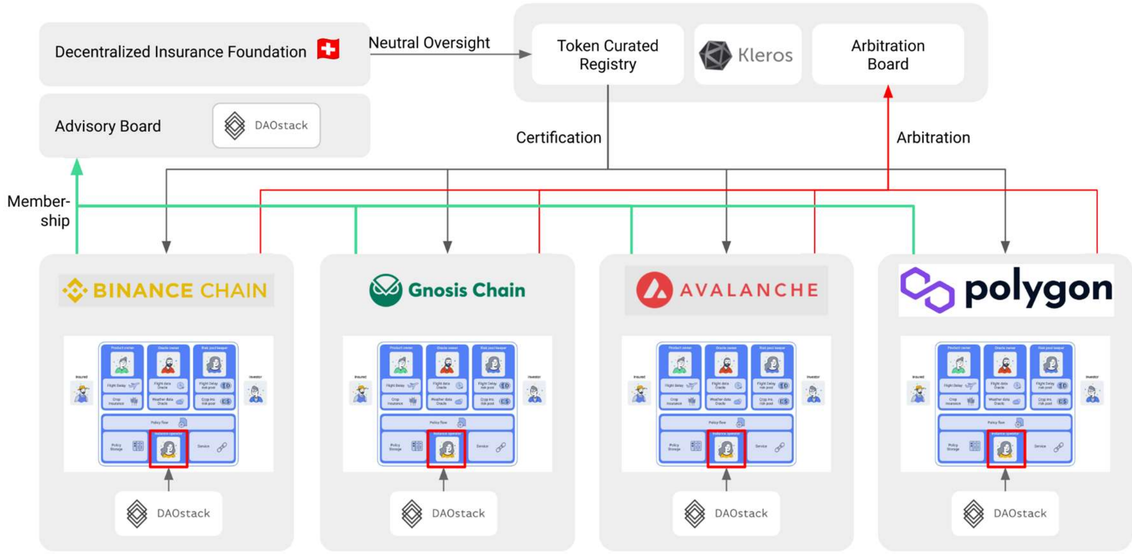
3. **Responsibility**
   Each participant shall act fully on his/her own responsibility, while the platform will provide any means to support this. All participants acknowledge their joint responsibility for the operations and development of the platform as a whole.
4. **Trust**
   The platform encourages trustful behavior and will provide a safe environment for all participants. Each participant is committed to compliant behavior. Transparency is an important element in trust-building, therefore we encourage transparency as much as possible, without violating the justified needs for protection of each participant of the platform.
5. **Public good / Commons**
   The platform as a whole serves the public good. It is a "commons"[15] in the sense of Elinor Ostrom and operated by the community of all participants. Therefore, the governance rules for the platform are based on the eight rules for successful commons, coined by E. Ostrom. In chapter 5, we discuss how the "eight rules" are implemented in the EGM and DIP Protocol.

5.3 High Level Structure of the EGM

In the image below a number of actors/participants are mentioned, the names mentioned are written as an example and it may be that other actors and/or blockchains are added. In this image the following players are mentioned;

Name	Short description
Decentralized Insurance Foundation	Development and promotion of the DIP-protocol, funding of the development of the Generic Insurance Framework (GIF)
Kleros	Decentralized arbitration service and Token Curated Registry
DAOstack	Software stack for DAOs including a library of governance protocols and interfaces for creating and managing DAOs
Binance Chain	Example Blockchain
Gnosis Chain	Example Blockchain
Avalanche	Example Blockchain
Polygon	Example Blockchain

[15] https://en.wikipedia.org/wiki/Elinor_Ostrom; see also https://www.onthecommons.org/magazine/elinor-ostroms-8-principles-managing-commmons



1. The four defining aspects of the EGM are as follows:
    a. Platform participants as the topmost authority
    b. The Decentralized Insurance Foundation as the non-profit, neutral link to the real-world institutions and legal systems
    c. Certification of GIF instances as a market signaling mechanism to incentivise high standard of work
    d. Dispute resolution via an independent arbitration board
2. The participants of the platform - be it insureds, product builders, or investors - are the topmost authority of the platform. Their stake is represented by governance tokens (vDIP), which are minted against staking DIP tokens in a governance contract. Governance tokens (vDIP) are used for decision making in all DAOs involved in the platform.
3. While the participants of the platform are represented by addresses on blockchain protocols, we need a link to the real world connecting the on-chain infrastructure with legal entities in the real world.
4. In the real world ("IRL"), the topmost authority is the not-for-profit Decentralized Insurance Foundation (DIF), based in Zug, Switzerland, and regulated according to Swiss Law.
5. The purpose of the DIF is defined in the notarial deed of the Foundation and cannot be changed:
   "The Foundation's purpose is promoting and developing new technologies and applications, especially in the fields of new open and decentralized software architectures mainly in the insurance field. A dominating but not exclusive focus is set on the promotion and development of the so-called DIP-protocol

and the related technologies, as well as the promotion and support of applications using the DIP-protocol."

6. Therefore, the only purpose of the Foundation is to serve the community of participants in building and using the DIP protocol.
7. The DIF is committed to strict neutrality. Therefore, the DIF will never engage in disputes between participants. For dispute resolution, the DIP Platform will use existing mechanisms like e.g. the Kleros arbitration board.
8. The DIF is formally represented by the Foundation Council.
9. The main task of the DIF in the context of the technical DIP Protocol is the certification of GIF Instances on the different blockchains. On each blockchain, there can be multiple GIF Instances.
   The rules for certification will be published. The rules should be such that, if possible, there is no ambiguity in interpretation and that people with basic technical understanding and common sense can make a decision whether a particular GIF Instance meets the requirements.
   Requirements include technical stability (like contract audits) and soundness, as well as legal compliance.
   Certified GIF Instances are registered in a Token Curated Registry, initially operated by Kleros. Should it turn out that Kleros ceases operations or that there are other strong reasons to change the registry provider, the DIF can change the provider.
   The concrete rules for certification of GIF Instances are currently work in progress.
10. Certification has no specific consequences - it's just signaling "this GIF Instance has undergone thorough scrutiny and due diligence and it implements the rules and recommendations of the DIP Protocol". Thus, we expect that a certification will act as a strong differentiator in the market and non-certification will essentially be a "red flag" for both customers and investors. This is how self-regulation works.
    However, in the future, other parties outside the DIP ecosystem could link access to certain services to valid certificates.
11. Each GIF Instance is operated by an Instance Operator. An instance operator can be represented by an EOA (externally owned address), a multisig, or a DAO. It is recommended that the Instance Operator is represented by a DAO, the members of which are the stakeholders of this GIF Instance.
12. Each GIF Instance may send a delegate in the Advisory Board of the DIF. The advisory board shall interact with the Foundation Council and represent the interests of the GIF Instances and its stakeholders with the Foundation Council. The advisory board and its decision-making processes are implemented as a DAO.
13. Each GIF Instance (or the DAO representing it) can implement governance rules on a more granular level, e.g. rules to decide which products may be listed on

the instance and which not, as long as these rules are in accordance with our core values and the other rules of the platform.

14. Each GIF Instance needs to implement rules which ensure that the instance is able to participate in the funding of the EGM and the DIP protocol in general.
15. Disputes are resolved via an arbitration board, initially hosted on Kleros. Possible disputes include e.g. registration of a GIF Instance in the TCR, or disputes in relation of insurance claims which cannot be resolved via smart contract logic (e.g. oracle malfunction).

5.4 Funding of the EGM and the DIP Protocol

1. The infrastructure to maintain the EGM, as well as the development and maintenance of DIP protocol (especially the development and maintenance of the GIF Framework) requires funding.
2. The funding is intended to only cover costs, to be self-sustaining and not profit-oriented.
3. Each GIF Instance will therefore be required to:
    a. Stake a defined amount of DIP tokens in a governance staking contract (for details, see 5 Global Staking Pool)
    b. Pay a regular fee to cover the operational cost of the EGM
4. The required amount of stakes and fees are calculated based on the economic volume which is transacted on the particular instance.
   The exact scheme will be published in time.
5. In case of violation of rules, sanctions of different severity can be applied to misbehaving participants:
    a. Financial penalties for misbehaving members
    b. Slashing of staked DIP tokens
    c. Exclusion of participants from a GIF Instance
    d. Exclusion of a GIF Instance from the Token Curated Registry.
6. Part of the fees paid will be burned to create a slight deflationary effect on the DIP token.

5.5 Global Staking Pool

1. The DIF will maintain a global staking pool (GSP). The GSP will be deployed on the Ethereum Mainnet and it will comply with the ERC900 Simple Staking Interface.
2. The GSP has the following objectives:
    a. Provide an economic incentive for well-behavior
    b. Provide a “sink” which will bind DIP tokens
    c. Ensure that participants which profit from the Etherisc ecosystem, have “skin in the game” and aligned interests with the whole system.

3. Participants in the Etherisc ecosystem are expected to stake and lock a certain amount of DIP tokens in the GSP:
    a. GIF Instance operators need to stake and lock tokens for each certified GIF Instance
    b. Product owners need to stake and lock tokens for each product deployed and approved on a certified GIF Instance.
    c. Oracle providers need to stake and lock tokens for each oracle deployed and approved on a certified GIF Instance.
    d. Risk Pool Keepers need to stake and lock tokens for each risk pool deployed and approved on a certified GIF Instance.
    e. Staking in the GSP is independent from staking in Risk Pools. Investors can stake in Risk Pools without having staked in the GSP, and the rules defined in this chapter do not apply to staking investors.
4. The amount to be staked and locked for each group of participants will be published on the Etherisc Website.
5. The amount to be staked and locked will correlate with the economic value which is created by the participant. The exact KPIs to be considered and the formulas to calculate the amount of DIP tokens to be staked will be published on the Etherisc website.
6. Tokens staked in the GSP can be locked by the stakers for different purposes:
    a. For a GIF Instance (necessary for operation of a GIF Instance)
    b. For a product (necessary for operation of a product)
    c. For an oracle (necessary for operation of an oracle)
    d. For a risk pool (necessary for operation of a risk pool)
    e. For specific governance purposes (optional, to participate in specific governance decisions)
7. For each purpose, there is a "Lock Manager" who has the power to lock or unlock tokens. Initially, the lock managers are controlled by a multisig owned by the Foundation Council of the Decentralized Insurance Foundation. After a testing period, the control on the lock managers may be transferred to the DAO associated with the Decentralized Insurance Foundation.
8. Each participant who has staked and locked DIP tokens will be granted general voting rights in the Etherisc Governance Model. For specific purposes, there may be the requirement to additional stake and lock tokens in a governance lock manager. For each governance decision, the voting rights are calculated on a snapshot of the GSP at a certain block height.
9. Voting is performed by Snapshot voting using a strategy which reads the locked tokens out of the GSP at a certain block height.
10. The code of the GSP is published in the global staking repo in the etherisc github.

5.6 Monetary policy of the DIF

1. As a major holder of DIP tokens (about 70% of the total supply of DIP tokens), the DIF is obligated to protect the interests of the DIP token holders. The treasury of the DIF is not counted in the circulating supply of DIP tokens.
2. The DIF may allocate grants or provide DIP tokens to incentivize the development and use of the DIP protocol. These grants and incentives will increase the circulating supply and could therefore lead to a dilution of the value of the DIP token. However, the DIF will always take care that grants and incentives are always in relation to the value created, so that the DIP token in total does not experience unnecessary dilution.

5.7 Appendix: Eight Rules for successful commons and how they are implemented in the DIP Protocol

3. Commons need to have clearly defined boundaries. In particular, who is entitled to access to what? Unless there's a specified community of benefit, it becomes a free for all, and that's not how commons work.

   The "boundaries" are implemented by the token curated registry for the GIF Instances, and the registries for products, oracles and risk pools in the GIF Instances itself.
4. Rules should fit local circumstances. There is no one-size-fits-all approach to common resource management. Rules should be dictated by local people and local ecological needs.

   The rules are always created on the lowest possible level. E.g. the top-level rules only govern which GIF Instances are certified. More granular rules are implemented on lower levels and they can be different for different GIF Instances, according to their needs.
5. Participatory decision-making is vital. There are all kinds of ways to make it happen, but people will be more likely to follow the rules if they had a hand in writing them. Involve as many people as possible in decision-making.

   Participation is implemented by the DAOs which govern the GIF Instances. Each GIF Instance is a member of the Advisory Board of the DIF and can represent their interests there.
6. Commons must be monitored. Once rules have been set, communities need a way of checking that people are keeping them. Commons don't run on good will, but on accountability.

The monitoring happens on two levels: The top level is given by the DIF, the Token Curated Registry of GIF Instances and the Arbitration Board. On a lower level, the monitoring is given by the DAOs governing the individual GIF Instances.

7. Sanctions for those who abuse the commons should be graduated. Ostrom observed that the commons that worked best didn't just ban people who broke the rules. That tended to create resentment. Instead, they had systems of warnings and fines, as well as informal reputational consequences in the community.

   There are different methods of sanctioning, each with a different level of severity, see chapter 3.
8. Conflict resolution should be easily accessible. When issues come up, resolving them should be informal, cheap and straightforward. That means that anyone can take their problems for mediation, and nobody is shut out. Problems are solved rather than ignoring them because nobody wants to pay legal fees.

   This is implemented by the arbitration board which offers dispute resolution on every level.
9. Commons need the right to organize. Your commons rules won't count for anything if a higher local authority doesn't recognise them as legitimate.

   This is implemented by the written rules which govern the DIF and which in turn govern the DAOs representing the different GIF Instances.
10. Commons work best when nested within larger networks. Some things can be managed locally, but some might need wider regional cooperation – for example an irrigation network might depend on a river that others also draw on upstream.

    This is implemented by the hierarchical structure, the top of which is a legal foundation recognized by Swiss law.

6 Abbreviations and Glossary

[t.b.d]