Acorn Protocol: Open Collaboration Protocol Based on a Trustless Foundation

Labor is humanity's oldest, most important method of value production. Classical economics recognizes labor as a primary element of production in addition to land and capital. According to the labor theory of value, labor is the only thing that can produce “value”. Labor plays an extremely important role as a component of the economic system as the starting point of value production.

Most individuals living today are creating value and confirming their existence through the labor market. Yet aside from the importance of labor, advancement of the labor market and labor contracts/organization methods have been stunted due to legal, fiscal, and accounting expenses. Information asymmetry based on nation, language, region, and individuals, inefficient legal contracts, issues with managers/mid-level supervisors during employment retention/evaluation processes are all contributing factors to overall inefficiency in the labor market.

We hope to organize large-scale labor forces in a more efficient manner and approach the topic of creating a global labor market based on a trustless foundation in protocol form in which manpower can be exchanged as a product.

This protocol will enable large-scale collaboration between both employers and workers, as well as among workers themselves, regardless of location or time. It allows for trustless collaboration on a large scale and will resolve the issue of trust cost in the verification process.

This topic includes work verification, which measures the output of labor exchanges, large-scale labor force operations made possible by automation and scaling, as well as protocolizing the labor market network. Operation standardization and subdivision, various verification mechanisms, and evaluation/distribution strategies all must be implemented to achieve these goals. Blockchain and its foundational technologies greatly improve efficiency and cost issues throughout the course of implementing these strategies.

Table of Contents

1. Problems with the Current Labor Market

1.1. Labor Market Differences Between Nations and Regions

The formation of global supply chains, advancement of information and communications technology, and proactive movement via immigration etc., have all contributed greatly to the integration and unification of the global labor market to a certain point. However, legal, fiscal, accounting, and other differences in infrastructure policies have caused fragmentation between labor markets amongst nations. As a result, inefficiency has been a recurring point of contention, consequentially generating issues with unemployment and opportunity costs.

More recently, global supply chains have been collapsing due to certain groups inciting “division” in order to exploit the situation and gain political advantages. Immigration has been being denied and the number of various opportunities for collaboration have been declining. Gaps between labor markets have increased and inefficiency is rapidly increasing in the overall market.

1.2. Information Asymmetry Throughout Employment Processes

The most foundational problem with the employment process lies with the information asymmetry arising from the disparity between job descriptions provided by the employer versus the workers’ actual skill sets.

Employers evaluate workers using indirect criteria that coincide with the position such as academic background, experiences etc., and ultimately base their hiring decisions off of these factors. Various conditions and qualifications are utilized during this process as pretext for efficiency. Information gained via interviews are also only indirect pieces of evidence to evaluate the candidate’s work performance capabilities. These indirect elements may or may not be relevant to the position. workers also judge the relevance between their skill sets and the position based on job descriptions, as well as work environment details, compensation system etc. communicated throughout the interview process. Yet most decisions made throughout the hiring process are subjective.

Despite having gone through such an extensive, costly hiring process, it is highly likely that decisions made may not be the best option for both employer and worker. Of course, this is only under the assumption that the hiring process concluded without any hitches. Employment is usually unsuccessful despite numerous applications, interviews, and negotiations. The expenses incurred throughout this process are costly.

1.3. Inefficiency Behind Labor Contracts

Labor contracts are legal contracts between employers and workers with respect to duties and conditions. Due to the fact that all labor contracts are accompanied by a series of expenses, most are renewed or discarded after they have been evaluated, and even then, only after a set amount of time has passed. The gap in time between the initial drafting of the contract, time of execution, and the time of completion gives rise to inefficiency.

Labor contracts are formed based on various pieces of information collected before the drafting stage. It is difficult to reflect changes that occur after finalization of the contract. workers are committed to performing labor within the scope of the contract and pursue optimization, ultimately eliminating the need for additional performance improvement. There are no factors that encourage workers to achieve beyond conditions/objectives transcribed on the labor contract. Employers also have no reason to provide additional compensation or actively adjust wages based on improved skills or productivity. Employers are hesitant to make wage adjustments that reflect real-time productivity increases because it is difficult to re-lower wages based on decreased productivity levels.

Most contracts stipulate restrictions such as minimum labor hours, minimum workloads, minimum payment units, etc. because entering and maintaining labor contracts require expenses. The absence of such limitations may result in a rapid increase in incidental expenses. Aforementioned restrictions put constraints on the number of active individuals in the labor market, subsequently limiting the employers' pool of choice.

Labor contract signing, renewal, and release is accompanied by various costs in most countries. These are mostly a byproduct of government policies that are meant to protect workers' rights and interests. Such policies result in an inert or overworked existing workforce for employers.

1.4. Work Verification and Inspection Costs

Employers verify workers' compliance with the labor contract upon signing the contract. They supervise the labor processes personally or hire a separate manager. The act of employing workers in itself is accompanied by management duties that evaluate the acts of labor, as well as verify the product of labor.

In this day and age of industrialization, the most rudimentary management procedures involve a manager estimating production units by the hour, coming up with a production goal, and operating in accordance with that goal. At the same time, various forms of evaluation methods and management tools are needed to ensure productivity rates by the hour. Distinct processes must be implemented in order to estimate the quantity and quality of labor.

Increase in volume of employment also means expansion of management processes. This increase forces the creation of various hierarchies in management and labor divisions. As a direct consequence, numerous management expenses are rapidly incurred. The appearance of various divisions makes accurate evaluations of labor commitment and production difficult and gives way for qualitative evaluations such as “teamwork skills”. Additional higher tier supervisor positions are now needed to evaluate the middle manager.

Yet another inefficient step is added as the supervisor is given the authority to adjust working conditions and begins to provide qualitative analysis on the worker. Office politics arise as discrimination in hiring, maintaining, and managing personnel occurs due to various reasons such as gender, age, race, etc. This is inconsistent with the employer's original objective (normally profit maximization). An agency-principal problem arises where the employer and manager's profit maximization no longer coincide with each other. This goes beyond efficiency problems in employer operations but leads to overall inefficiency in the entire labor market.

2. The Need for a New Protocol

Labor has been branching off in various ways following the advancement of information and communication technology, as well as unique internet based industries. And it also became possible to express and store the newfound processes and outcomes derived from labor in data format. This in turn signifies that various business tasks are initiated and completed with networks at its forefront, and that more forms of business tasks can be exchanged online in the form of services.

Some professional services have already begun to appear on the online market in the form of outsourcing. The worldwide pandemic was also an opportunity for many to confirm that working remotely was viable for more jobs than we thought possible.

workers in the form of digital nomads have been on the rise for certain industries where they work while traversing across various nations and regions. These nomads are occasionally met with situations where their employer's country, the country they are residing in, and their country of citizenship are all different. Labor laws, tax and accounting systems, as well as wage payment methods are unable to properly handle situations like these.

From an employer's perspective, the emergence of various digital workers leads to increases in work verification and trust maintenance costs between the employer and worker. Existing labor contracts have lost their original meaning, and established hiring and management processes have also begun to lose significance.

Employer (corporation) operation and sales methods have begun to change significantly in the wake of labor format changes. Unlike how businesses normally would initially expand their head office then branch out into smaller local offices, many companies are now forming their business models around a cloud base and directly reaching out to the global market. Companies are hiring employees from various nations in an outsourcing format, or in the form of a remote work environment. Employing and managing various workers from various countries has complicated company operating procedures and in turn, has increased management costs. This issue has become one of the biggest obstacles for the growth of cloud-based companies.

It has become quite evident that there is a need for a network based open labor supply and demand market that is not restricted by a specific country or region. This new network must be in a format where both employers and workers can easily access it. The relationship between employers and workers must not retain its current labor contract form but be transparent and efficiency focused. workers will be fairly compensated through this new contractual relationship and employers will also be able to improve system productivity and efficiency. This new system must be developed in protocol format in order to ensure expandability, as well as interconnectivity between various tasks. This protocol will enable large-scale collaboration between both employers and workers, as well as among workers themselves, regardless of location or time. It allows for trustless collaboration on a large scale and will resolve the issue of trust cost in the verification process.

3. Protocol as the Infrastructure of the Labor Market

3.1. An Overview of the Acorn Protocol

The objective is to manifest a global labor market upon a network base. Accomplishing collaboration between various people spread across the globe based on a trustless foundation. Our aim is to develop a protocol where employer demand is defined, acts of labor can be verified, and compensation/value preservation can be realized.

The inception of this protocol may lie with altering existing labor contract formats into a smart contract. Blockchain smart contracts are transparent programs that display the predetermined conditions of an agreement where when “a certain condition” is met, “a certain outcome (compensation)” is executed. The “condition” and “outcome” cannot be arbitrarily changed after finalization. As long as an individual is compliant with the “condition” anyone can receive the “outcome” specified in the smart contract. If this were to be applied to labor contracts, a more open yet outcome based and efficient contractual labor relationship could be established, resulting in a significant decrease in management expenses.

This new infrastructure must be developed in the form of protocol. This protocol must be open, interchangeable, and interoperable. Anyone who utilizes this protocol must be able to develop a platform in which various jobs from various industries can be accomplished. Anyone should be able to provide their labor services to the global labor market and receive equitable compensation in return. This new technology must allow all individuals to be able to directly participate in a larger global economy without any physical, regional, or associative restrictions. employers must be able to scale their services at a low cost and utilize reasonable labor in an efficient manner.

A mechanism that can make decisions on a trustless foundation must be systematized and automated for this protocol to be employed at a large scale. It must also fragmentize job details so that anyone can understand the content and so that tasks can be quickly distributed. It must verify the outcome of the task in various ways and provide immediate feedback. Rapid feedback and awarding of compensation/penalty based on results can lead to an increase in worker and company productivity. This will ultimately lead to a virtuous cycle in which an increase in employer competitiveness leads to increased demand for workers.

3.2. Series of Process Execution

3.2.1 Standardized Microtasks

Labor expands across a wide array of tasks from simple physical jobs to complex cognitive assignments. The knowledge and skill set threshold for task completion and inspection becomes higher as work processes become more and more complicated. Compensation for the job and verification both rise proportionately as well. More time is required, and the probability of misunderstanding and misjudgment increases. Convoluted tasks require divide and conquer strategies.

They can be divided and distributed until the task becomes a standardized microtask. Upon completion, the outcome of these microtasks can be knitted back together to take up a form in which the employer's demand is met.

Simplified microtasks lowers the participation threshold for workers and reduces training expenses. Many more individuals that are able to provide simple labor services will be able to simultaneously work on the project, resulting in quicker processing speeds. Verification processes also become faster and more simplified. It is much simpler to go through a single criteria of verification rather than through multiple criteria simultaneously. Initial development and training costs can be reduced through task standardization by utilizing overlapping tasks. Task-related skill and credibility evaluations can be cross-referenced amongst workers working on the same project.

The quantity of tasks also rises as complicated tasks are split into simpler tasks, while the unit price of each simpler task is lowered. This may lead to expense increases in project management, accounting, and expense settlements. Blockchain, however, will be able to significantly lower these execution costs.

3.2.2. Inspection of Intersections Between Independent workers

Acts of labor can be bifurcated into new tasks and verification processes. All workers participate in the entirety of the process as both the issuer and verifier. It is difficult for the worker to differentiate between the two. They also cannot choose to partake in a specific format of labor. Everyone is simultaneously both a worker and verifier.

The product created during the process of newly issued work must be verified by a non-biased third party. The worker must be anonymous when the product is distributed to verifiers and verifiers must follow a criterion when carrying out their duties. The verified

product may also be redistributed to another unbiased third party for additional verification. Dispersion systems in place between workers, new workers and verifiers throughout the process of labor distribution allows for independence in each role so that all parties involved can remain impartial.

The platform may create an additional verification step or issue a penalty to the initial worker upon detection of an issue with the product. A mechanism in which either compensation distribution is locked, or retrieved in the case of those already distributed. Verification duties may also be reevaluated and considered as an initial task. In the event that a malicious action is identified during the verification process, a penalty may also be applied, locked up, and retrieved.

The employer may also choose to not automate verification processes and either personally verify or commission the task to another individual. In such instances, the employer must provide all workers with precise feedback in a timely manner. In the event that a certain time frame has passed, the employer is renouncing any claims of product rejection and/or application of penalties to the worker.

3.2.3. Workers and the Credibility System

The workers' account wallet plays the role of a decentralized identifier (DID). Workers that are represented by account wallets can build credibility by utilizing past projects and verification reports. Protocol and platforms may refer back to previous tasks when evaluating the worker's skill sets, technique, and attitude. Tasks and costs required for verification can be reduced in reference to this system. The saved expenses can then be redistributed to workers. The worker can maintain their accounts on a long term basis to maintain their standing in the credibility system to receive more opportunities and ultimately receive higher compensation.

As the worker builds up their credibility, the overall volume of verification tasks required via protocol will decrease and average worker profit will increase. Task distribution may occur in differing methods depending on the type of project. Each project and/or task may differ in required skills and techniques. Frequency and extent of verification may also differ as well. Higher standards raise the threshold in which workers must jump over, resulting in higher costs.

3.2.4. Abuse Prevention Mechanisms

Abuse prevention mechanisms are put in place so that workers cannot present random products or intentionally cause internal disruption via malicious submissions. Among these mechanisms, differentiation tasks serve to discern whether or not all workers are committed to the given task in a professional manner with good intentions by distributing select tasks that the platform already knows the answer to. Differentiation tasks are perfunctorily ordinary in both task and compensation. Thus, normal workers will not be able to distinguish them from others. Workers that are unable to complete the differentiation task will receive a penalty and be required to provide additional verification for all future tasks.

Abuse prevention mechanisms are based on the validity of the product submitted by the worker, but do not judge whether the submission was made via algorithm and/or bot. In other words, if a specific process within the project can be accomplished more efficiently via bot, then its participation is allowed. workers can participate in the protocol by developing and maintaining the bots.

Humans can perform different roles where human cognitive abilities are better suited, while bots can contribute in areas where overall efficiency improvement is needed.

3.2.5 End Product Review

Employers are able to conduct a sample inspection and/or total inspection depending on the needs of the project. Re-inspection processes and strategies applied to the final product are contingent upon the employer's decision. Employers can carry out a full-scale inspection on new types of projects/tasks at the early stage, or decide to go with smaller sample size inspections for more sophisticated processes and reduce costs. Discovery of erroneous, faulty products may result in additional feedback for the initial worker and verifier and be used as reference data within the credibility system

3.2.6. Ins and Outs of the Network

The most elemental function of all parties involved is that everyone directly produces and verifies products through a constant, controlled network. However, platforms can improve and edit products to meet the specific needs of the employer through the use of external Oracle databases and by referring back to external institutions'

verification content. This allows for flexibility and customization in the production and verification process.

Heterogeneous networks, or projects that occur outside the network frame can be verified through indirect methods using contingent data evidence. Inspection can also be carried out through external agencies and/or API. Results can be recalled by the work distribution system and credibility system.

End products adopted by the platform are ultimately products of consensus between new workers (issuers) and verifiers. Protocol enables systematic objectivity by making the process anonymous and more decentralized. This is a socio-technical product and does not differentiate between specific networks, both on and offline. If direct/indirect evidence on the product and its objective verification were to exist, a series of strategies could be utilized to record and validate various types of work results in homogeneous networks, heterogeneous networks, and out-of-network environments. The entirety of the process could be recorded within the blockchain and utilized in establishing additional strategies.

3.2.7. Governance via DAO

Decentralized Autonomous Organization (DAO) suggests alternative options for scenarios where corporations are disbanded and replacement options for labor unions and board of directors. Any and all participants of the protocol are registered as members of the DAO protocol and can partake in the operation of the protocol. The product is created in accordance with everyone's best interests in mind.

Definite lines of differentiation between employers and workers may very well disappear as a result. Everyone will be able to transparently participate in system operations based on each individual level of contribution, ultimately removing the concept of managers and agents. Some may say this is similar to that of Marx's 'Free Association of Producers'. Blockchain technology enables a number of individuals to participate in governance at a low cost and high efficiency. Protocol participants can engage in decision making via staking and voting

1) Selection of projects/tasks managed by protocol and review of compensation standards
2) Modification of penalty criteria for abusive workers

3) Linkage with new special purpose platform and/or removal of pre-existing special purpose platform
4) Long-term project direction designation
5) Governance token (OAK production) process amendment
6) Terms involving ecosystem funded subsidy

4. Protocol Application Platforms

Employers can carry out tasks by selecting required projects and tasks via protocol platforms and charging equitable compensation. In principle, project and task compensation can be set up by the employer, but only under the premise that it has been approved through the DAO. Types of jobs that can be carried out via native platforms can be continuously added through the DAO.

Employers may also develop and link special purpose platforms in order to distribute workloads more efficiently for workers and reduce long-term costs associated with task completion. Like so, overall completion costs can be cut down by reducing overall task difficulty by segmenting special duties within specialized industries into standardized jobs and providing relevant tools.

4.1. Linking with Data Labeling Service SaaS

4.1.1 Business Overview

The demand for training data rises in proportion with demand for machine learning technology. However, data labeling is extremely labor intensive, and economies of scale are not easily attainable. Most data labeling is enabled through multi-step outsourcing in developing countries. Various forms of information asymmetry exist within multi-national collaborations and the middleman has been taking home a good chunk of profit.

The market has been growing at a fast pace, but most growth occurs in the form of local businesses. The majority of demand surveys and sales processes have been completed through negotiations. Business expansion requires great amounts of fixed costs, essentially making it difficult for businesses to scale.

4.1.2. Overall Structure Blueprint

We provide a web-based SaaS platform that provides cloud-based data labeling services. Enable expression of data labeling demand as a combination of standardized tasks. Allow easy visualization of the entire process by transforming the combination into a workflow. Customers will be able to upload data that needs to be labeled and designate a fitted labeling demand based on a standardized tool. Upon receiving payment for the labeling service, the platform will distribute tasks befitting each phase using protocol. workers following protocol will complete their desired tasks and carry out verification procedures. The platform will review task results and convert the products into a complete label to be delivered to the customer.

In order for this to be made possible, a website that can define the customer's demand, a labeling tool that workers can easily access, and a statistical tool that can review and manage the products is needed. Unique types of tools can be developed to complement the various labeling demands. Development of labeling tools in mobile application format will enable more participation from workers, radically increasing the amount of work completed.

The platform will be able to charge a stable coin at the time of purchase. Expenses can be distributed in liquid token format to workers in real time via simple swap as the project is being completed. The accounting portion of the expense settlement process can be simplified even more through these procedures.

4.1.3. The Role of Protocol

Employers can start up a labor-intensive business without workers using the protocol. There is no need for the employer to directly hire any workers, nor do they need to pay any fixed expenses to establish corporations across the globe. Thousands of individuals will be able to simultaneously operate on a single project through task fragmentation and distribution processes, rapidly satisfying the needs of the consumer. Employers will be able to quickly respond to a rise in business demand and expand as needed. Internal production rates and costs will be predicted much more accurately through the fragmentation and standardization of processes. In turn, it will become possible to present a standardized billing system to consumers. Management, accounting, and remittance costs can be greatly reduced by handling the overall process through blockchain technology.

4.2. Feasible Applications

4.2.1. The Application of Automation Processes

Robotic process automation (RPA), spam filtering solutions, and other various automation processes have continued to integrate algorithms and humans alike. Despite the fact that many aspects are being taken care of through algorithms and/or other machine learning based solutions, many factors still require manual inspection via human cognitive abilities. It is possible to combine worker inspection processes by linking the protocol for processes that have low certainty factors. Employers will no longer need to directly hire workers and manage verification processes but simply pay in accordance with the volume of data processed. Revised data can be recycled as training data.

4.2.2. Global Promotion Campaign

Promotions agencies will be able to promote specific brands, products, and/or ideas simultaneously across the globe. Individuals attending the promotion will perform actions specified by the employer in specific areas and provide photographic proof that the action was completed. Once an independent third party verifies the validity of the worker's work, compensation is paid out. Employers will be able to restrict participants based on their geographic location, language, age, etc., as well as time restrictions. Employers will be able to receive indirect evidence on the product and provide said evidence as output.

4.2.3. Design Draft Contest

Employers will be able to host a design draft contest targeting designers across the world. Precise concepts and requirements for the design will need to be stated, as well as compensation for participation. Participants will be able to utilize tools of their choice to create various design drafts. The appearance of generative AI will allow for more people to express their creativity as a designer. Employers will be able to amass a large volume of drafts within a short span of time. All participants that can meet the criteria set out by the employer will be able to get compensated. The employer will participate in the final selection process to choose the draft to be used while the selected designer will receive greater compensation for their work.

5. Technical Considerations

5.1. Overall Technical Structure

The objective is to help the understanding of overall development strategies and basic structures of protocol and platform creation by using data labeling businesses as an example. The overall structure is designed with interchangeability in mind so that support goes beyond specifically formatted jobs such as data labeling and onto a wider spectrum of tasks.

The overall software required to make protocol a reality is composed of three layers. 1) Base platforms include elementary functions of the task (overall task flow) and finance related functions (accounting and wages). 2) Link modules that connect task distribution strategies, evaluation strategies, and external strategies independently from the base platform are included. 3) Various tool applications that can support specifically formatted jobs and Oracle platforms that can securely deliver external resources exist.

5.2. The Structure of Each Individual Solution

5.2.1 Foundational Resources

Basic resources signify data structures that are used for tasks. This is the most elementary data format package that can be used regardless of the functions of the task. This data structure can already be considered as an abstract geometric structure. Basic resources can serve as both task input and output data. For example, if the basic resource were to package an image file, it could include the original image, compressed image, and image size/meta data. Word data packaging may involve including various translated versions of the text.

```
type Image struct {
    OriginUrl string
    CompressedUrl string  //compressed quality, image size before compress
ion
    Width int64
    Height int64
    ...
}

type Text struct {
    Origin string
    MultiLang map[int]string  // Multilingual Response
}

type Point struct {
    x float64
```

```
    y float64
}
type Points []Point
type Line   []Point
type Polygon []Point
type Rect   [4]Point
type RotateRect struct {
    Rect Rect
    Angle float64  // median line clockwise PI calculation 0 <angle<PI/2
}

// 3D Point Cloud related resource type
type Point3D [3]float64
type Size3D  [3]float64

type DBox struct {
    Center Point3D
    CubeSize Size3D
    Quanternions [4]float64  // Quanternions, change to Euler angle or axi
al angle
}

type DPCD struct {
    OriginUrl string
    RotateUrl string  // After rotating pcd url
    FilterUrl string  // Filtered object after rotation, pcd url
    FloorHeight float64
    Quaternions [4]float64 // Quanternions
    ...
}
```

5.2.2 Basic Tasks

The structure of this layer must possess interchangeability. Unrestricted connection between basic tasks must be possible at any time. Data must be able to be transferred between the various tasks.

This structure can only be seen as complete when each task can encompass independent base resources and its products. Base resources and their products can have a 1: N relationship. For example, a single image can simultaneously be labeled as a box and polygon.

The task in itself must be separated from its functions from the overall process' standpoint. The task's functions could be to directly create the product, to evaluate the worker, or to test abuse, but the function is independent from the task's structure.

The task's structure is also irrelevant to the distribution of tasks. Tasks are not linked directly to the process. We've implemented the concept of task formats such as formal duty, abuse discernment, skill set detection by adding a new layer on top of the pre-existing layer. Upon completion of the task, each format is responded to with a different strategy.

```
// Basic task settings, essential resource for special format task source,
 user work format, production format setting

const (
    MultiPoints = iota
    FixNumPoints
    SingleRect
    MultiRect
    SinglePolygon
    MultiPolygon
...
}

type ActionSettings struct {
    ActionType int
    RequiredResource map[int]int // Required Resource
    OptionalResource map[int]int // Optional Resource
    Result int // Result format
}
type ActionResult []interface{}

// Basic task related information
type Action struct {
    ID string // Sole task ID, irrelevant to resources and related only to
tasks
    ActionType int // ActionSetting type
    Resource map[int][]interface{}
    Result ActionResult
}
```

5.2.3. Types of Tasks

This format is only relevant to tasks currently being executed and completely independent from former or latter tasks. It is also irrelevant to processes. This task format includes specific task information related to user, reward, and decision strategies. This task format has a one-to-one correspondence with the aforementioned basic task.

```
// User related information
type User struct {
    ID string
    ...
```

```
}

// Reward structure
type Reward struct {
    AcceptAmount int64  // Reward
    RejectAmount int64  // Penalty
    FinalAmount int64  // Output Reward and Penalty combined, can possess both positive and negative numbers, can be renewed
}
type Task struct {
    ID string        // task ID must be present
    TaskType  int    // task type
    Action Action
    User User
    Status int       //  0 Pre-distribution, 1 Post-distribution, pre-work, 2 Post-distribution, pre-completion, 3 Completion
    StdAnswer ActionResult. // Standard Result, Action.Result must be of the same type
    EvlStrategy *EvaluateStrategy    // Evaluation strategy for this task
    Reward Reward     // Verification Standards and Results
    UpdateTime int    // Time Renewal
}
```

5.2.4 Process

Processes denote the primary execution structure of a project. Each project and process structure form a one-to-one response relationship. Processes are composed of combinations of nodes. Each node utilizes actions as its base, not tasks. Tasks are formulated only when the overall process, resource, user, and strategy is combined throughout the distribution process. From there, they are distributed to users. This structure in and of itself cannot form a complete workflow nor a circulation structure. This structure cannot fulfill a complete workflow.

Processes simply show the flow of data while distribution mechanisms transfer actions in task form to determine how the tasks will be distributed to which worker. Specific node tasks within the process are not distributed to users and may be transferred directly to the next node. This is also determined by distribution strategies.

```
// The relationship between nodes are determined. FullData and mapping relationship between node action resources are included in addition to data link forms
```

```
type Link struct {
    Source *Node
    Target *Node
    Mapping map[int]int    // Node.FullData, Mapping relationship of rear
node action resources
}

// Include another action. Nodes are also irrelevant to task types
type Node struct {
    ID string
    Action Action
    Count int     // volume of responding base resources, specific nodes
can Complete actions simultaneously
    Distribute *Distribute   // Distribution strategy
    FullData map[int]interface{}   // Does not signify specific data nor
all types of data including recent nodes
}

// Project settings
type ProjectSetting struct {
    IsSkipWorkBox bool
    ...
}

// Project
type Project struct {
    ID string
    Setting ProjectSetting
    RootNode *Node // Must be unique
    ResultNode *Node // Must be unique
    Nodes []*Node
    Links []*Link
}
```

5.2.5 Task Distribution Strategies

Task distribution strategies decide task volume and sequence. The distribution system distributes tasks to users in advance to maximize efficiency. Users selected based on an evaluation strategy system are pulled from a pool to be assigned tasks in advance so that workers can promptly participate in various tasks. Tasks that have gone past a certain date, tasks that have been completed by all users assigned to the task, and tasks that are triggered by specific conditions are sorted through the distribution system so that

non-completed tasks can be redistributed to other users. Redistribution is repeated until the task is completed.

```
// outcome processing strategies, omitted if distribution number is 1.
A method must be selected if greater than 1 or is the exact number, the
one of the following must be selected: sum of sets, intersecting sets. D
efined by user

type ProcessStrategy interface {
    DoneTask(task *Task)
   }

 // distribution strategies, evaluation strategies must be included
type Distribute interface {
    SetEvlStrategy(strategy EvaluateStrategy) error // evaluation strate
gy includes user pool,user profile, evaluation duty mechanisms
    SetDistributeCount(num int) // default 1 unit of distribution,if 0,
then process in accordance with outcome process strategy
    SetProcessStrategy(strategy ProcessStrategy) error
    DoneTask(task *Task) error // ProcessStrategy.DoneTask and Evaluates
Straategy.DoneTask is used to process
     }
```

5.2.6 Evaluation Strategies

Evaluation strategies exist as an independent system and do not rely on task systems or projects. User information such as credit score and work experience can be deduced by compiling all task outcomes using the evaluation strategy system. Distribution strategies can simultaneously refer back to evaluation strategies provided by various users. All outcomes that have been completed by workers are consulted by the evaluation system regardless of type.

```
// Evaluation strategies are included in all tasks. Post-processing is
executed after task completion. And all strategies are included within the
project process node's distribution strategy

// work ability, language, basic task ability related evaluation
type Capability struct {
```

```
    Language map[int]float64   // language is evaluated on a 0~1 scale
    Action  map[int]float64   // each ability based on different tasks
}
// Statistical work record data, 1 month, 1 week, 3 days, data on the day
of
type Record struct {
    TaskCnt                // Time range
    CorrectRate float64     // Overall valid task ratio
    AvgTaskCnt []int        // Average throughput
    ActionTaskCnt map[int]int   // Throughput per task
}

// User management
type Worker struct {
    User *User   // Platform related user information, generally only
requires ID
    Capability Capability   // Various evaluation information on users
    Recods map[int]Record   // Format, Total Volume, 1 month, 1 week or 3
days int response
}

// Strategy formats where any user can participate
type EvaluateStrategy interface {
    GetID() string
    GetName() string
    GetDescription() string
    AddWorker(user *User) error // Addition of new users to the worker
pool
    DelWorker(userID string) error
    GetWorker(userID string) (*Worker, error)
    GetNext(projectID string) (*User,*Task, error)
    UpdateCap(userID string, sign int, score float64) // Skill set ability
 update
    DoneTask(userID string, taskID string, sign int) // sign 0- default 1-
correct 2- not correct
    AddAction(projectID string, action *Action)
}
```

5.2.7 User System

Among task systems, user systems already define structure when it comes to user related data. However, this is mostly defined using task files. The user system discussed here is usually utilized to record information unrelated to the task. The system contains information drafted directly by the user and/or information that needs to be verified by the platform, or information that does not need to be verified. Another function of the

user system includes organizational access to protocol user-related information on the platform. Platforms can take care of financial matters directly related to the platform workers through this function.

The user system's key role involves assigning all users a status, verifying their status using protocol, and granting authority over all actions carried out via protocol. The user system in and of itself does not individually evaluate status or hand out penalties. An example of what happens is that when a specific user carries out a task but causes an abuse issue, the evaluation strategy applies a penalty in conjunction with rectifying that user's evaluation. This evaluation can also only be carried out under authorized parameters. The distribution strategy system rearranges the distribution strategy in accordance with the user's requests and original strategy, while the user system does not partake in this process. Even under severe infractions, the penalty and managerial action administered for abuse by the evaluation strategy system cannot go beyond restricting users from participating in any further tasks and is unable to remove the user from protocol.

The platform may charge fees during the worker registration process or collect additional, partial expenses from the work reward. These additional expenses may be helpful in maintaining user status on platforms and driving users to put more emphasis on their status.

```
CreateUser() (privateKey string) // Create User, include user's wallet
address etc

// User system basic structure
type User interface {
    GetAddress() string
    Login(privateKey string) error
    Logout() error
    GetInfo(public bool) map[string]interface{}
    UpdateInfo(key string, value interface{}) error
    JoinES(es *EvaluateStrategy) error
    GetESInfo() ([]interface{}, error) User evaluation based on different
strategies
    // Does not matter if task data is derived from internal platform data
Only utilized within the platform
    GetTaskList() []string, error
    PullTask(ID string) *Task, error
    PushResult(task *Task) error
```

```
    DepositeFund() error
    PendingFund() int64
    AvailableFund() int64
}
```

5.2.8. Various Support Tools

Employers can provide suitable work tools as needed in web and mobile format. Employers can verify identity as well as connect to protocol via Software Development Kit(SDK) and/or wallet verification methods. Self-review platforms can be developed in order to inspect and combine. Employers can provide additional feedback on task outcomes. Strategy and user systems can refer back to this. Employers can produce new outcomes or data by fusing protocol outcomes, self-data, and/or external data. Employers can develop a self-regulating system in order to review overall protocol productivity, as well as improve the process. All support tools presented here can either be added or omitted based on user demand.

5.3. Application of Blockchain Technology

Overall protocol process strategies convert strategically complicated tasks into simple, standardized tasks, resulting in increased work volume. Tasks also see an expansion in the number of types, including actual tasks, verification tasks, etc., provoking a larger volume. At the same time, workers of various backgrounds are working on various tasks 24 hours a day. Throughout that process, blockchain technology reduces the cost of its overall development processes, as well as operation and accounting/settlements expenses.

Task processing phases are recorded in conjunction with data combined with reward by workers and verifiers. This is an outcome created through general processes and may be utilized for evaluating future product issues. Lockup mechanisms of the token enable users to efficiently render penalty mechanisms.

Task volume rapidly increases as complicated tasks are segmented into standardized, simple tasks. At the same time, as the global labor market's various participants complete assigned tasks, the current blockchain solution may still see issues with regards to payment punctuality and associated costs. As a result, the protocol must

be developed alongside the continued use of high-performance blockchain networks and expansion to various networks.

Our long-term vision is to see protocol ported onto various networks so that more users can be connected with developers and reach abundant liquidity. Various cross chain and network expansion solutions may additionally be utilized to achieve this goal.

6. Token Economics

6.1. Dual Token System

The Acorn Protocol has established a dual token system in order to better differentiate between liquidity provision within the labor market and governance. It is composed of Acorn (ACN), which serves as liquidity rewards acts of value creation within the Acorn protocol. It is also composed of Oak (OAK), which is responsible for overall protocol governance.

6.2. Acorn (ACN)

Acorn (ACN) is Acorn Protocol's primary liquidity token. Protocol users and workers utilize ACN for expenses. All projects must precisely estimate task unit prices and the overall budget, as well as charge ACN, which amounts to the entire estimated budget. Even when the entire budget is charged in stable coin form, the protocol swaps the stable coin out for ACN throughout the project and pays out the worker in ACN. Protocol governance token OAK can only be produced by fusing ACN, and in that process, ACN is mostly incinerated.

A maximum of 10,000,000,000 ACN can be issued at once. 27% of the total issued ACN has been distributed to blockchain holders, 8% to team operation rewards, 20% to ecosystem establishment funds, and 20% to foundation operation related costs. The last 25% serves as a subsidy for Acorn and will be used for distribution purposes for initial service creation.

6.3. OAK (OAK)

Both employers and workers can participate in DAO governance through OAK within the Acorn Protocol. Debate on the overall direction of the Acorn Protocol via Acorn DAO can be found in section 3.2.7. Specific terms about governance via DAO can be voted on.

A maximum of 120,000,000 OAK can be issued at once. OAK has no initial issuance and can only be produced by fusing ACN. OAK can be produced by fusing a certain number of ACN, can be produced per every 28,800 blocks (defined as 1 round), and is limited in the production quantity per round. The volume of issued OAK will be regulated based on time of project development in addition to adjustment of the amount of ACN required to produce OAK. The total amount of OAK produced per phase and the amount of ACN needed to produce said OAK is as follows.

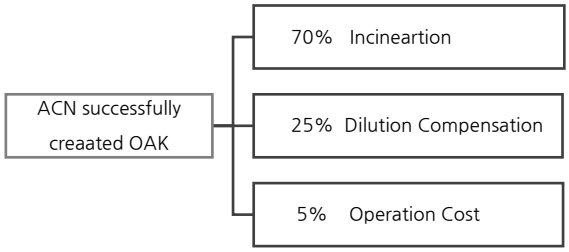
	Pre term	Term 0	Term 1	Term 2	Term 3	Term 4	Term 5
Round * Frequency	15	30	180	180	180	180	360
Maximum OAK per round Productivity Volume	1,000,000	500,000	250,000	125,000	62,500	31,250	15,625
Quantity of ACN required to produce 1 OAK	10	20	40	80	160	320	640
Maximum accumulated OAK production	15,000,000	30,000,000	75,000,000	97,500,000	108,750,000	114,375,000	120,000,000

* A single round is equal to 28,800 blocks on the MARO Blockchain. MARO blockchain produces 1 unit of block per 3 seconds and 28,800 blocks corresponds to approximately 24 hours.

Anyone who possesses a wallet filled with ACN can participate in the production process of OAK. In the event of successful OAK production, most of the ACN used is incinerated, while some are distributed to current OAK holders as dilution compensation, while others are paid out as protocol operation fees.

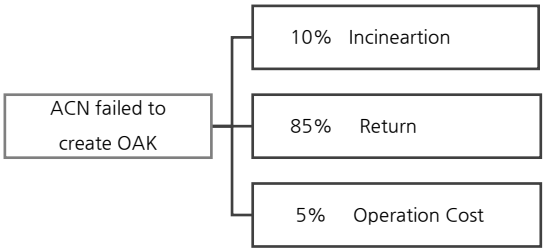
When the amount of ACN needed for OAK production per round is less than the required amount, the amount of OAK produced that round is determined by the volume of ACN. OAK that was not produced during the completed round must be carried forward to the next round of production. As such, the final amount of OAK production may be less than what is theoretically possible.

ACN disposal after initial OAK production is set up as such, and proportions are altered based on polls.



The majority of ACN that partook in OAK production processes but were unsuccessful due to daily production restrictions are returned to the

participant while a small amount is incinerated.



OAK holders can receive dilution compensation through staking and can also participate in various rectification issues on new OAK production through polling.