



CAT.TRADE

CAT.trade

Decentralised Oracle Network for Market Data

Lightpaper v1.0 (August 21st 2020)

By Keith Wareing, David Lyons, Steve Hall, Tristan Pretty and Tim Biddiscombe

Abstract

The advent of smart contracts has the potential to revolutionise the construct of future businesses. The fixed nature of smart contracts means that dynamic data that continues to fluctuate cannot interact with the blockchain without the introduction of a new functionality known as an *oracle*.

In this paper, we aim to summarise how a decentralised oracle structure can enhance the decentralised retail and algorithmic trading experience by adding a 2nd layer to the Ethereum blockchain which interacts with aggregated centralised market data and commands. The aim is to provide modular elements of missing functionality to decentralised exchanges in a fast and trustless manner that is not hindered by excessive gas fees.

We explain how we will structure a decentralised node network by using a gas free public blockchain protocol that interacts with our ERC20 CATX token to address several issues we have identified across the decentralised exchange sectors whilst developing our own advanced algorithmic tools for use with centralised exchanges, and how the harvesting of market data for our centralised tools has led us to this solution.

1. Introduction
2. Spot Trading Tools
3. Orderbook Visibility
4. CATX Protocol
5. Oracle Nodes
6. Marketplace
7. CopyCAT
8. Token Distribution

1. Introduction (a brief history of CAT.trade)

Let's first commence with a brief summary of our achievements and products.

CAT.trade was born from humble beginnings which stemmed from the desire to automate a specific indicator script. The initial script evolved from its 1.0 version to its current 3.0.1.8 version which has been fully optimised for trade automation via the popular leverage exchange Bybit.

CAT which stands for Crypto Algo Trader is cloud based, and utilises the popular Telegram platform as its interface, which has allowed us to add extra functionality to the trading experience on our partnered exchange which include but are not limited to.

- Entry signals (e.g. webhooks from Tradingview) for trade entry and exit via CAT settings such as take profit, trialling stop, indicator condition exit.
- Quick Fire Entry & Exit via Webhooks
- CopyCAT indicator and Strategy Presets (allowing users to simply copy successful strategies and traders)
- Manual Trading with additional conditions (such as instantly open with specific exit conditions) direct from your phone
- Post Trade P&L reporting, including trading fee breakdown so you actually know exactly what you are spending per trade, allowing traders to pinpoint areas where fees can be limited

Since the launch of CAT in April 2020 over \$2.8bn has been traded through our platform, averaging \$600m per month.

2. Spot Trading Tools (Centralised Exchanges)

CAT.trade is now nearing completion of our second development iteration, to bring spot trading tools to Binance & Kucoin which will see us begin to move away from Telegram to a new web interface, whilst still retaining some use of Telegram for instant mobile alerts.

The integration of our first 2 spot exchange partners, identified as Kucoin and Binance, have been selected based on a combination of factors such as wide variety altcoins to trade, high liquidity, and orderbook depth.

WebCAT will be hosted directly on CAT.trade and it's first function will be to automate the buying and selling of assets in and out of stable coins using our algorithms and indicators.

3. Orderbook Visibility

One problem that retail traders face on centralised spot exchanges is Orderbook Visibility. The moment you place an order on a centralised exchange it's left in plain sight for all to see.

This can be particularly problematic if you are trying to sell a large altcoin holding, as you end up bidding against other bots, traders, and market makers.

With CATs Buy The Dip bot, you can set large buy and sell orders completely hidden from the order book, once the desired price is reached CAT will execute a market sell or buy. CAT will use the current orderbook data to determine how much of the order can be filled, and repeat the process as price is reached until the order is filled.

Create BOT

1

Exchange & Pair

Learn more about all our connector something

2

Strategy

Learn more about all our connector something

3

Start Condition

Learn more about all our connector something

4

Profit

Learn more about all our connector something

5

Safety

Learn more about all our connector something

Target Profit

2.5 %

Type

Percentage from total volume

☒ Trailing

Deviation

0.25

PREVIOUS

NEXT

Overview

Name	My Demo BOT
Exchange	Binance
Pair	BTC, ETH, LTC, etc
Strategy	Long
Target Profit	2.0 Percent, Average, etc
Deal Start Condition	Q1, Position Trader...
Base Order Size	0.00017771 BTC
Safety Order Size	0.000216158 BTC
Max safety trades count	3
Max active safety trades count	1
Price Deviation	25
Start Order type	Limit
Initial order volume scale	
Safety order step scale	
Stop Loss %	
Stop Loss action	
CoolDown between deals	
Minimum price to open deal	

Traders can add hidden ladders to DCA in to a trade, and the same can be done for exits.

However one advantage of hiding your orders from the exchange orderbook is that you can use one piece of capital on several orders.

Dashboard
CAT Trading
My BOTS
Exchanges
Portfolio
Marketplace

Create BOT

+ Connect an Exchange

Exchange

Binance

Nickname

Binance Bitcoin

API Key

API Secret

Connect

Cancel

Period

Binance

For example:

User has 1btc on an exchange and sets an order to use the full 1btc to buy ETH at \$300

On a spot orderbook, that 1 bitcoin is locked and cannot be used for anything else.

With CAT that order can be placed, but whilst you are waiting for the order to fill, you can choose to buy and sell in and out of tether automatically based on an algo or indicator.

Or you could set low buy orders on every single asset on that exchange. Whichever order hits first would be filled.

It is this functionality such as this that has led us to looking at a fully decentralised model.

4. CATX Protocol

In order to provide our tools in a decentralised fashion, there first needs to be a gas resistant alternative to communicating with Ethereum based smart contracts.

This is where we introduce our CATX Protocol, where traders and developers can deploy unique actions via a 2nd layer on the Ethereum blockchain, such as adding buy orders or sell orders at a specific price on any global market via oracles, whilst giving users the option to hide this information from the public or chose to share it to earn residual income paid in our native CATX token.

Take for example the rise of Uniswap, a great decentralised exchange that is growing in popularity.

However the unique way in which uniswap works, means that sellers cannot set a sell order to trigger at a specific price, meaning traders that are limited on time, that do not wish to sit on the exchange watching the price, are forced back to the centralised exchanges in order to set such an order.

However, the promise of adding buy and sell orders to Uniswap is not as simple as it sounds. Ethereum smart contracts would have to constantly reference a price feed as an action on the Ethereum blockchain, which each time will incur a small gas fee.

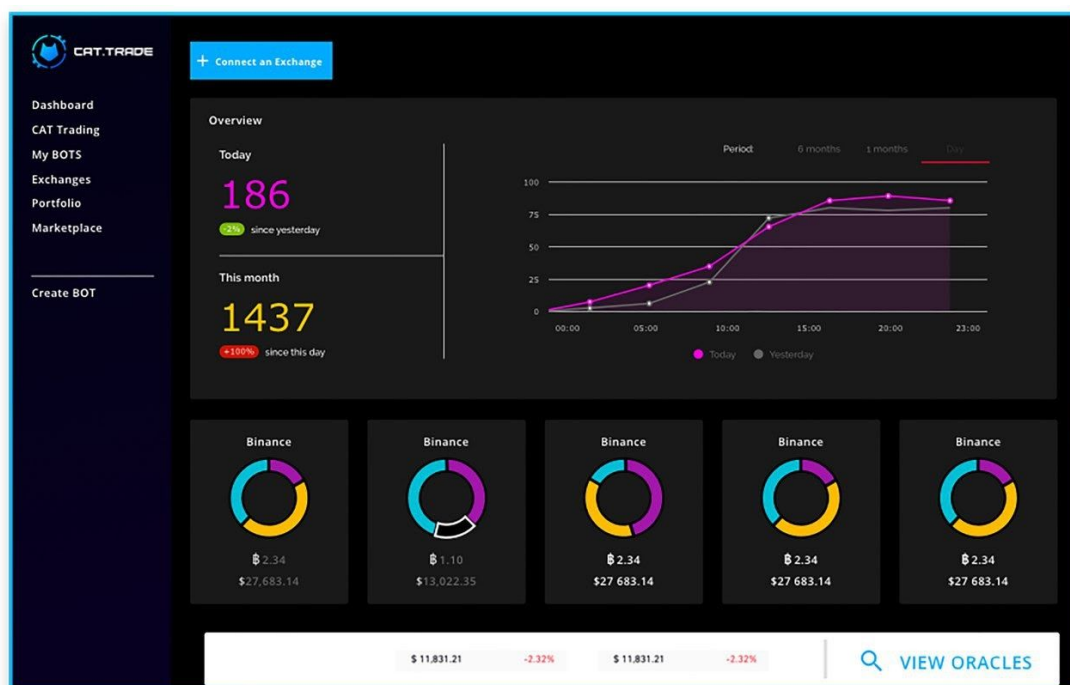
Extrapolate this fee over several minutes, hours or even days, you could exhaust your entire capital just in paying gas fees.

The introduction of oracles validated by on-chain nodes hosted on our public gas free blockchain solves this.

The CATX oracles will consist of 300 nodes each requiring a stake of 50,000 CATX tokens.

This accounts for 50% of the circulating supply of CATX, the remaining tokens will be freely tradable on centralised and decentralised exchanges to make them easily accessible to anyone that wants to purchase oracle data for their trading price feeds.

Market data is already gathered by CAT.trade for the purposes of algorithmic trading bots that can be built in a modular fashion via the CAT.trade algo builder for use on centralised exchanges.



We then aggregate the data from several sources for our oracles, but why trust just us? What if we got it wrong or decided we wanted to manipulate the price of a particular feed to fill an abundance of orders at well below market rate?

The solution is that once we pull the market data and aggregate it, it must first be validated by our oracle nodes, and only then our oracles can dynamically adjust the parameters that will form the smart contract.

So if an aspect of a smart contract can be changed, does this lose the element of decentralisation? The solution is for each node to provide validation that market data is correct.

This form of consensus amongst the initial 300 nodes operating on a second private blockchain separate to our erc20 token chain can be further pooled to smaller nodes for a wider consensus network as demand grows.

Node Oracles will receive equally 100% of the CATX payments for aggregated data.

The use case for this data adds huge functionality to cumbersome limited decentralised exchanges, and can also provide complete privacy for the retail trader. Information such as buy and sell orders and orderbook depth can be shielded from bad actors such as rogue market makers and whales to ensure maximum profitability can be achieved when trading on decentralised exchanges.

5. Oracle Nodes

CATX tokens required to run a node is set at 50,000 and no more than 300 initial nodes will be made available.

Nodes will validate the market data and the commands that the user defines on the private blockchain before sending it to the main smart contract on Ethereum to be executed.

For example:

User wants to sell 5 ETH at a price of \$500 USDT on Uniswap allowing x% slippage

The oracle will only send this action to the main chain at the point the price is reached and after all nodes have verified that data and action.

The flow and pricing differs depending on whether the oracle creator wishes for the order to be private or public.

Public

User creates order via oracle. Costs 4 Units of CATX (not necessarily 4 whole CATX, it will be relative to the current price of gas so that the order can be executed on the ETH chain).

- 1 unit is converted to ETH for gas
- 1 unit is used for creating the order (paid to nodes)
- 2 units are used for executing the order (paid to nodes)

If the order never executes and is cancelled

- The 1 unit converted to ETH for gas remains locked
- The 1 unit paid for creating the order stays with the nodes
- The 2 units are returned as a rebate to the user

The user creates another order

- 1 unit is again paid to the nodes
- The gas unit isn't needed because it's already there
- 2 units are locked for execution, and paid to nodes if order is executed, and rebate to user if order is cancelled.

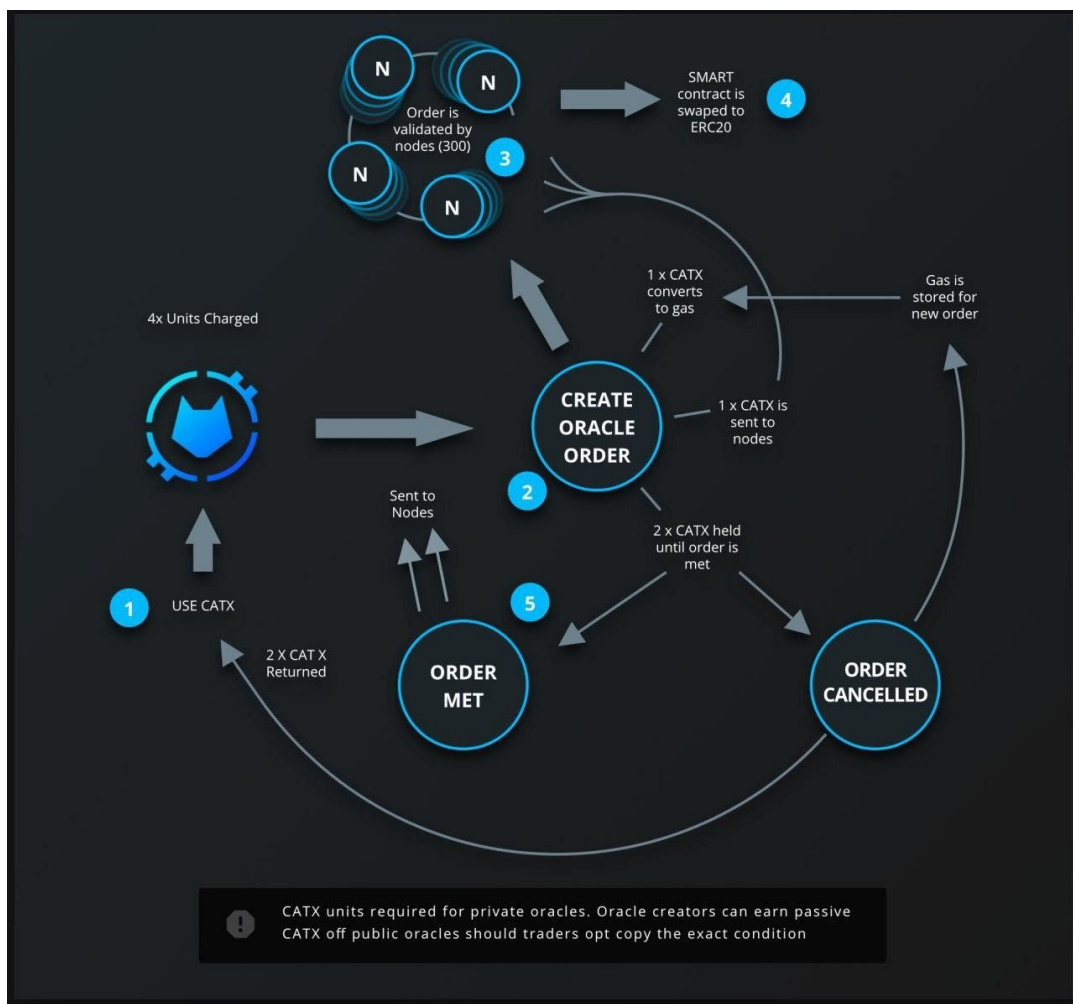
However if another user wishes to copy this oracle order, they only pay 3 CATX units, two at point of order one of which is converted and added to the gas fee the other to the oracle

creator and one at point of execution. In this scenario the opening CATX unit is paid not to the node.

This allows people creating oracles to receive a reward should people copy their condition, and provides the buyer with the advantage of taking part in a larger public position. This is particularly useful when it comes to DeFi as smaller investors tend to be out priced by high gas costs.

Should the order be cancelled 2 x CATX units are returned, and the gas fee remains locked in the contract.

Under this structure it naturally drives the demand for CATX whilst simultaneously providing relative reward to those running oracle nodes and creating public orders that are visible for all to see.



Private:

User creates order via oracle. Costs 8 Units of CATX (not necessarily 8 whole CATX, it will be relative to the current price of gas so that the order can be executed on the ETH chain)

- 2 units are converted to ETH for gas
- 2 units are used for creating the order (paid to nodes)
- 4 units are used for executing the order (paid to nodes)

If the order never executes and is cancelled

- The 2 units converted to ETH for gas remains locked
- The 2 units paid for creating the order stays with the nodes
- The 4 units are returned as a rebate to the user

The user creates another private order

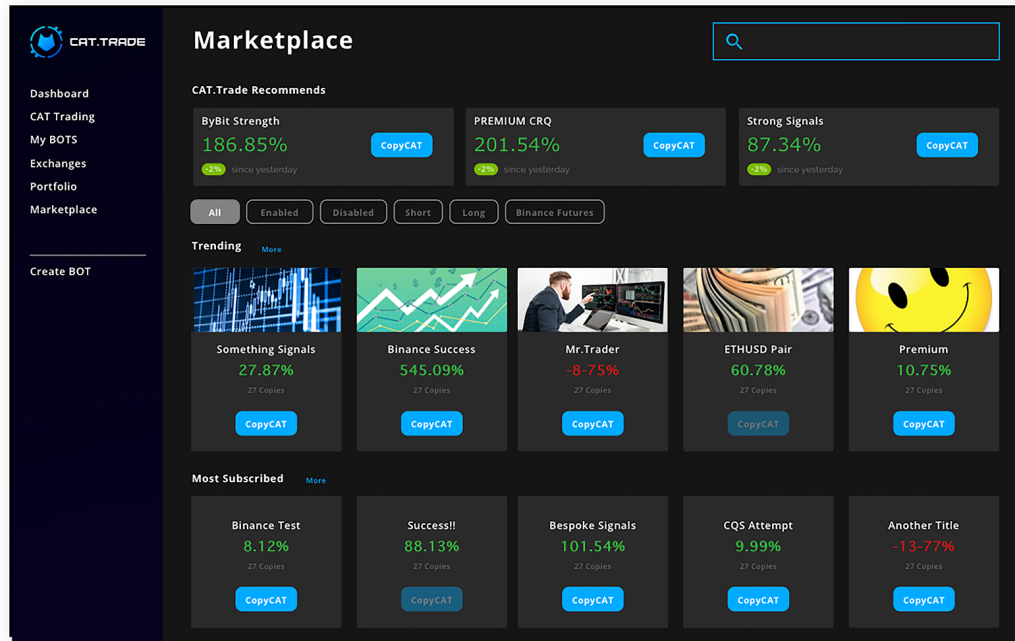
- 2 units are again paid to the nodes
- The gas unit isn't needed because it's already there
- 4 units are locked for execution, and paid to nodes if order is executed, and rebate to user if order is cancelled.

Under this structure node holders receive greater compensation for ensuring that condition of the oracle is made private, so that others cannot front run a trade.

The structure of public oracles will be the building block for native dApps on the CATX blockchain that can be constructed on the CAT.trade platform and made available in the CAT.trade marketplace.

6. Marketplace

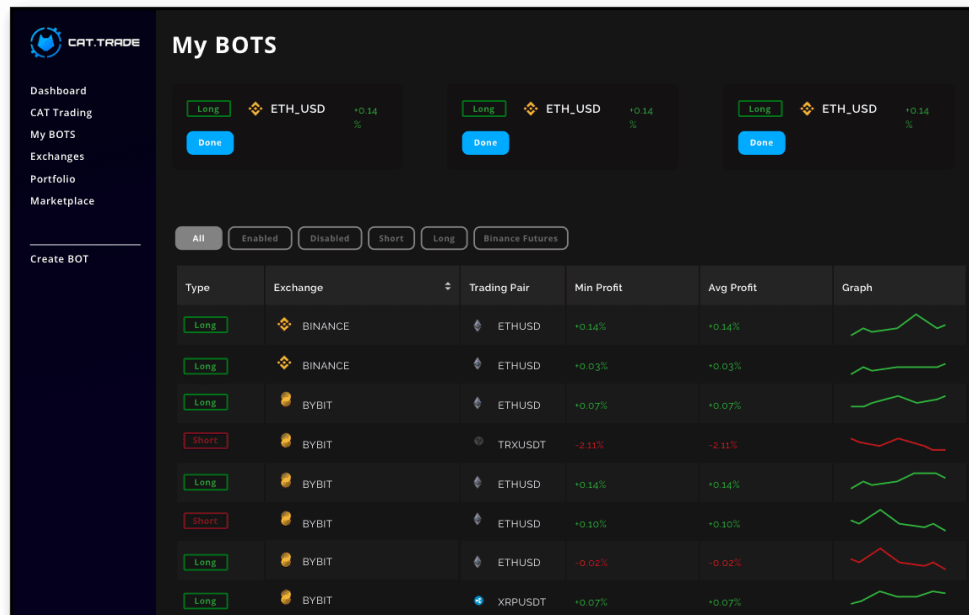
An integral part of CAT.trade, the Marketplace is made up of several elements.



- Indicator marketplace where people can sell their own indicators that can be automated with CAT on Spot and Leverage exchanges
- Instant Affiliate protocol where sellers can embed a CAT payment widget that will split incoming payments into seller and affiliate wallets at point of sale. Initially this will be for BTC payments with the option to add further payments assets including CATX at a later date
- Oracle marketplace where people can create algorithms powered by CATX oracles and build and sell custom trading bots, or simply create one off orders on specific decentralised exchanges. Such as the ability to add buy and sell orders to Uniswap
- Node marketplace where the initial 300 nodes can add sub nodes under their structure. Here if a node holding 50,000 CATX wishes to reduce their holding, they can do so by listing a swap at market rate based on CATX/ETH determined by our Oracles to add smaller node holders as the network grows and further bandwidth and computational power is needed for the oracle node blockchain, to ensure SLAs are met and maintained.

7. CopyCAT

One of the key features of CAT.trade will be the ability for traders to simply copy an algo, bot or any form of trading action that a user makes public.



This will be compiled as a live leaderboard displaying the most profitable traders and algos, with a simple “few clicks and live trading” ethos being applied so that users can easily replicate another traders success.

Whilst the majority of features on CAT.trade will be free, there will also be certain premium features that users can subscribe to, however to access CopyCAT there will be a one-off payment in Bitcoin equal to just \$20 of which 100% is paid to the referrer.

This system is designed to provide all users an incentive to bring people to the CAT.trade platform, furthermore traders are incentivised with large Bitcoin prize pools for the top performing algo or trader on CopyCAT on a monthly basis.

The decision to use Bitcoin for this, and not CATX token is to keep the CATX ecosystem stable for oracle users until such time sufficient liquidity is established.

8. Token Distribution

Ticker

CATX

Total Supply

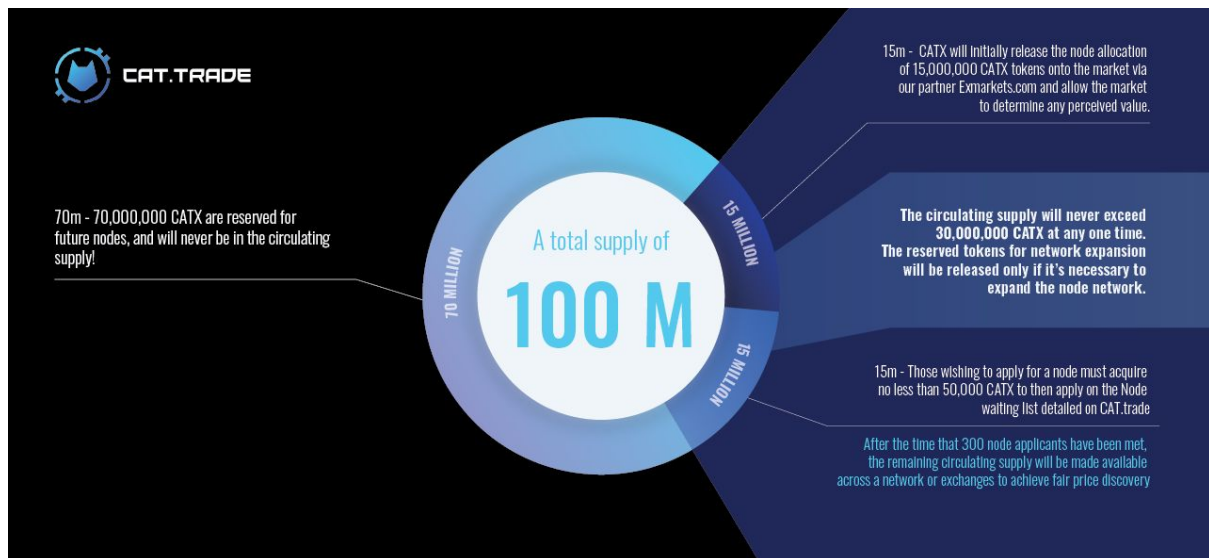
100,000,000

Circulating Supply

30,000,000

Reserved for Network Expansion

70,000,000



CATX will initially release the node allocation of 15,000,000 tokens onto the market via our partner Exmarkets.com and allow the market to determine any perceived value.

Those wishing to apply for a node must acquire no less than 50,000 to then apply on the Node waiting list detailed on [CAT.trade](https://cat.trade).

After the time that 300 node applicants have been met, the remaining circulating supply will be made available across a network or exchanges to achieve fair price discovery, and will be required when users wish to interact with an oracle.

The circulating supply will never exceed 30m at any one time. The reserved tokens for network expansion will be released only if it's necessary to expand the node network.