Sinar Chain

**مقدمه‌ای بر فانتوم**

فانتوم (Fantom) یک بلاکچین متخصص قراردادهای هوشمند است که در سال 2018 بنیان‌گذاری شد. توکن بومی آن FTM نام دارد. این شبکه تلاش می‌کند مشکل مقیاس‌پذیری (scalability) را که بسیاری از بلاکچین‌های قدیمی‌تر مثل اتریوم با آن دست و پنجه نرم می‌کنند، حل کند.

### ویژگی‌های کلیدی شبکه فانتوم

### ****مکانیزم اجماع (Consensus Mechanism):**** فانتوم از الگوریتم اجماع منحصربه‌فردی به نام **Lachesis** استفاده می‌کند. این الگوریتم بر اساس یک **DAG (Directed Acyclic Graph)** طراحی شده و از خانواده **aBFT Asynchronous Byzantine Fault Tolerant** است.

* **aBFT**  به این معناست که شبکه حتی در صورت خرابکاری برخی نودها هم می‌تواند به اجماع برسد.
* **Lachesis**  به دلیل طراحی DAG و موازی‌سازی، بسیار سریع و کم‌هزینه است.

**سرعت بالا و کارمزد پایین:**  
زمان نهایی شدن تراکنش‌ها (Finality) در فانتوم معمولاً **1 تا 2 ثانیه** است و کارمزد بسیار پایین (معمولاً زیر یک سنت!) دارد.

**EVM-Compatible:**  
فانتوم با ماشین مجازی اتریوم (EVM) سازگار است. یعنی می‌توان **اسمارت‌کانترکت‌های اتریوم** را به راحتی در فانتوم دیپلوی کرد.

**مقیاس‌پذیری بالا:**  
فانتوم برخلاف بلاکچین‌های سنتی، به هر dApp اجازه می‌دهد یک بلاکچین اختصاصی راه‌اندازی کند (مانند **Avalanche Subnets** یا **Polkadot Parachains** این باعث می‌شود هر dApp بتواند به صورت مستقل کار کند، بدون اینکه ازدحام سایر dApp ها تأثیر منفی روی آن بگذارد.

### معماری و اجزای اصلی فانتوم

**Opera Chain:**  
اصلی‌ترین بلاکچین فانتوم که از مکانیزم Lachesis استفاده می‌کند و با EVM سازگار است.

**Lachesis Protocol:**  
هسته اصلی اجماع است که تراکنش‌ها را سریع و ایمن تأیید می‌کند.

**FTM Token:**  
رمز ارز بومی شبکه. کاربردهای اصلی:

پرداخت کارمزد تراکنش‌ها

استیکینگ (staking) برای ایمن‌سازی شبکه

حاکمیت (governance) – رأی‌دادن به تصمیمات مهم شبکه

### کاربردهای فانتوم

**DeFi:**  
فانتوم میزبان بسیاری از پروتکل‌های دیفای مانند SpookySwap، SpiritSwap، BeethovenX و … است.

**NFT:**  
پشتیبانی از بازارهای NFT بومی و توسعه سریع در این حوزه.

**dApps و DAOها:**  
به دلیل سرعت و هزینه پایین، فانتوم میزبان پروژه‌های مختلفی از جمله سازمان‌های خودگردان غیرمتمرکز (DAO) است.

**مزایا و معایب**

**مزایا:**  
مقیاس‌پذیری بسیار بالا  
کارمزد پایین  
نهایی‌سازی سریع تراکنش‌ها  
سازگاری با EVM

**معایب:**  
 هنوز نسبت به اتریوم یا BSC ، کاربران کمتری دارد.  
 رقابت بالا با دیگر بلاکچین‌های مقیاس‌پذیر مثل Avalanche، Solana و ….

**جمع‌بندی**

**فانتوم** یکی از سریع‌ترین و ارزان‌ترین شبکه‌های بلاکچین است که با رویکرد DAG-based و الگوریتم اجماع Lachesis، مشکل مقیاس‌پذیری را به شیوه‌ای جالب حل کرده. با رشد DeFi و نیاز به تراکنش‌های سریع و ارزان، فانتوم جایگاه خوبی پیدا کرده و به یکی از گزینه‌های محبوب در اکوسیستم Web3 تبدیل شده.

## ساختار کلی پیاده‌سازی شبکه فانتوم

### هسته شبکه : Lachesis Protocol

**Lachesis** قلب تپنده Fantom است. این پروتکل یک الگوریتم اجماع مبتنی بر **DAG**  و از نوع **aBFT** (Asynchronous Byzantine Fault Tolerant) است.

#### **مفاهیم کلیدی Lachesis:**

**DAG (Directed Acyclic Graph)**: برخلاف بلاک‌ها در بیت‌کوین یا اتریوم، Lachesis داده‌های تراکنش را به‌صورت گراف جهت‌دار غیرمدور ذخیره می‌کند.

**Event Block** : هر نود به جای ایجاد بلاک‌های بزرگ، به‌طور مداوم **Event Block** می‌سازد که حاوی تراکنش‌های جدید، لینک به Eventهای والد (parents) و متادیتا است.

**OPERA DAG**: کل شبکه از روی Event Block ها تشکیل یک DAG به‌نام **OPERA DAG**  می‌دهد.

#### **اجماع Lachesis**

اجماع Lachesis در سه مرحله اتفاق می‌افتد:

1. **. Creation of Event Blocks**  
   هر نود به‌طور مستقل Event Block می‌سازد و اطلاعات زیر را در آن قرار می‌دهد:

هش والدین (parent events)

هش‌های رویدادهای قبلی

لیست تراکنش‌ها

امضای دیجیتال نود

1. **Gossip Protocol هم رسانی رویدادها**

**Gossip**: هر نود رویدادهای خودش و سایر نودها را به‌طور تصادفی به همسایه‌ها می‌فرستد (شبیه یک ویروس!).

این باعث می‌شود کل شبکه به‌سرعت از وجود Event های جدید باخبر شود.

1. **Finality و اجماع**

Lachesis از الگوریتمی شبیه **"فقط نیاز به دانستن"** استفاده می‌کند.

هر Event Block در نهایت شامل داده‌هایی می‌شود که آن را به اصطلاح **"Root"** یا **"Clotho"** می‌نامند.

در نهایت Eventی که در شبکه به‌طور کامل توسط همه تأیید شده باشد، به **"Atropos"** تبدیل می‌شود و تراکنش‌هایش نهایی (Finalized) می‌شوند.

| **🏷️** | **توضیح** |
| --- | --- |
| **Root** | رویدادی که اولین بار در یک round دور خاص ساخته می‌شود و اجماع در موردش سریع‌تر انجام می‌گیرد. |
| **Clotho** | ریشه‌هایی (Root) که در roundهای بعدی در Eventهای بسیاری ارجاع می‌شوند و اجماع در موردشان قابل‌اطمینان می‌شود. |
| **Atropos** | Clothoیی که نهایی شده و یک زمان قطعی (consensus timestamp) به آن اختصاص داده می‌شود. این رویدادها باعث نهایی‌شدن تراکنش‌ها می‌شوند. |

### مراحل اجماع Lachesis

**Round-based:**

* Lachesis شبکه را به **round**‌های متوالی تقسیم می‌کند.
* هر نود در هر round تعدادی Event Block می‌سازد.

**Fame Decision**  انتخاب رویدادهای معتبر:

* الگوریتم Fame Decision بر اساس رأی‌گیری غیرمستقیم (مثل **voting by gossip**) انجام می‌شود.
* Eventهایی که رأی‌های کافی بگیرند، معتبر (famous) تلقی می‌شوند.

**Ordering ترتیب‌دهی:**

* پس از انتخاب Clotho و تعیین Atropos، Lachesis ترتیب رویدادها را (topological ordering) در DAG تعیین می‌کند.

### اجزای فنی پیاده‌سازی

**OPERA Chain:**

* پیاده‌سازی بلاکچین یا دقیق‌تر: DAG-chain
* هر Event Block یک نود DAG است.

**State Database (Pruning & Storage):**

* برای ذخیره وضعیت (state) قراردادهای هوشمند، از پایگاه‌های داده‌ای مثل **LevelDB**  یا **RocksDB**  استفاده می‌شود.
* ساختار Merkle-Patricia Tree مثل اتریوم برای ذخیره‌سازی داده‌های وضعیت.

**EVM Compatibility:**

* فانتوم از EVM کامل پشتیبانی می‌کند.
* در لایه اجرای قراردادهای هوشمند، از کلاینت‌های **go-ethereum**  یا نسخه‌های سفارشی‌شده استفاده می‌شود.

**P2P Networking:**

* پروتکل‌های هم‌رسانی (gossip) معمولاً با استفاده از **libp2p**  یا مشابه آن در سطح P2P پیاده‌سازی می‌شوند.

### الگوریتم‌های دقیق‌تر در Lachesis

برای پیاده‌سازی Lachesis، الگوریتم‌های زیر کلیدی هستند:

**Gossip-based Synchronization**

* هر نود Event جدید را به‌طور تصادفی با همسایه‌ها به اشتراک می‌گذارد.
* این فرآیند در نهایت باعث تکمیل DAG مشترک بین همه نودها می‌شود.

**Root Selection & Root Graph**

* در هر round، رویدادهایی که برای اولین‌بار ظاهر می‌شوند، Root می‌شوند.
* DAG به Root Graph تبدیل می‌شود و این Root Graph مبنای رأی‌گیری است.

**Fame Decision Algorithm**

* مشابه **Snowball (Avalanche)** یا **virtual voting (Hashgraph)**:
  + هر Root در roundهای بعدی درباره Fame سایر Rootها رأی می‌دهد.
  + اگر یک Root به‌اندازه کافی رأی موافق بگیرد، به‌عنوان **famous** تأیید می‌شود.

**Clotho Selection**

* وقتی Rootی به‌اندازه کافی مشهور شد، Clotho انتخاب می‌شود.
* Clotho رویدادهایی است که به‌زودی به Atropos تبدیل خواهد شد.

**Atropos Selection & Consensus Time**

* Clotho بعد از roundهای رأی‌گیری، یک timestamp اجماعی دریافت می‌کند و به Atropos تبدیل می‌شود.
* Atropos تعیین‌کننده‌ی ترتیب نهایی تراکنش‌هاست.

### نهایی‌سازی (Finality)

* در Lachesis، Finality به‌طور قطعی (deterministic) است. یعنی وقتی یک Event به Atropos تبدیل شد، برگشت‌ناپذیر و دائمی است.

### مقیاس‌پذیری در عمل

* هر نود می‌تواند به‌طور مستقل Event بسازد (asynchronous creation).
* هیچ محدودیت سختی برای اندازه بلاک (مثل بیت‌کوین یا اتریوم) وجود ندارد.
* این مدل موازی‌سازی باعث می‌شود throughput شبکه بسیار بالا باشد (هزاران TPS).

## جمع‌بندی

**Lachesis DAG-based aBFT**: الگوریتم اجماعی سریع و مقاوم در برابر خرابی.  
 **Event-based و گراف غیرمدور**: به جای زنجیره‌ی خطی، یک گراف جهت‌دار غیرمدور ساخته می‌شود.  
 **سازگاری با EVM**: امکان اجرای قراردادهای هوشمند اتریوم.  
 **کارمزد پایین، نهایی‌سازی فوری، مقیاس‌پذیری عالی**: از دلایل محبوبیت Fantom.