NEWNAL DATA ARCHITECTURE

뉴날의 데이터 관리 구조

모으지 않고, 활용한다.

훔쳐 가도, 쓸 수 없다.

PERSONAL DATA OPEN SQUARE

PERMISSIONED DATA SPHERE

NEWNAL Inc.

목차

> **"너희는 왜 미성년 팬의 데이터가 안전하다고 말하는가. 우리와 무엇이 다른가." — 이 질문에 답하는 문서다.**

결론. 지금까지의 데이터 서비스는 운영 회사를 믿어야 성립했다 — 명부도, 권한도, 기록도 회사의 DB 안에 있었고, 회사만 볼 수 있었기 때문이다. 뉴날은 이 신뢰의 층을 회사 밖의 퍼블릭 블록체인으로 옮겼다. **소유 · 허락 · 접근의 기록이 누구나 확인하고 검증할 수 있는 공개 원장 위에 있으므로, 이 구조는 뉴날을 포함해 어느 회사에 대한 믿음도 요구하지 않는다.**

뉴날의 시스템에는 팬의 실명 · 주민번호 · 계정 · 전화번호 · 이메일이 저장되지 않는다. 암호화 키는 팬의 기기에만 있다. 그래서 외부에서 훔쳐 가도 쓸 수 없고, 내부에서 열어볼 권한을 가진 사람도 없으며, 뉴날도 팬의 허락 없이는 데이터를 활용할 수 없다. 그런데도 서비스의 모든 기능과 파트너의 데이터 활용은 전부 돌아간다. 이 신뢰의 층 노릇을 할 수 있는 블록체인이 하나뿐인 이유는 9장에, 그것이 이미 국가 규모로 실용화되었다는 사실은 10장에 있다. 아래는 이 구조의 방식이다.

---

01 유출이 사고가 되는 조건

> **유출이 폭탄이 되려면 세 가지가 한곳에서 만나야 한다: 내용 × 식별 × 도달.**

- **내용** — 민감한 정보(사연 · 감정 · 소비 · 건강).
- **식별** — 누구의 것인지(실명 · 계정 · 주민번호).
- **도달** — 그 사람에게 닿는 수단(전화번호 · 이메일 · 주소).

기존 서비스는 계정 때문에 세 항을 한곳에 모은다 — 회원 가입이 내용 · 식별 · 도달을 첫날부터 한 줄에 묶는다. 보안 투자는 이 문제를 해결하지 못한다. 방어는 언젠가 뚫리고, 데이터를 열 권한을 가진 내부자의 행위는 막지 못한다 — 실제 유출 사고의 상당수가 내부에서 일어난다.

뉴날의 시스템에는 식별과 도달이 없다. 훔쳐 가면, 누구의 것인지 모르고 누구에게도 닿을 수 없는 기록만 남는다. 쓸 만한 것을 얻으려면 팬의 기기를 한 대씩 뚫어야 하고, 한 대에 한 사람이라 경제적 효용이 나오지 않는다.

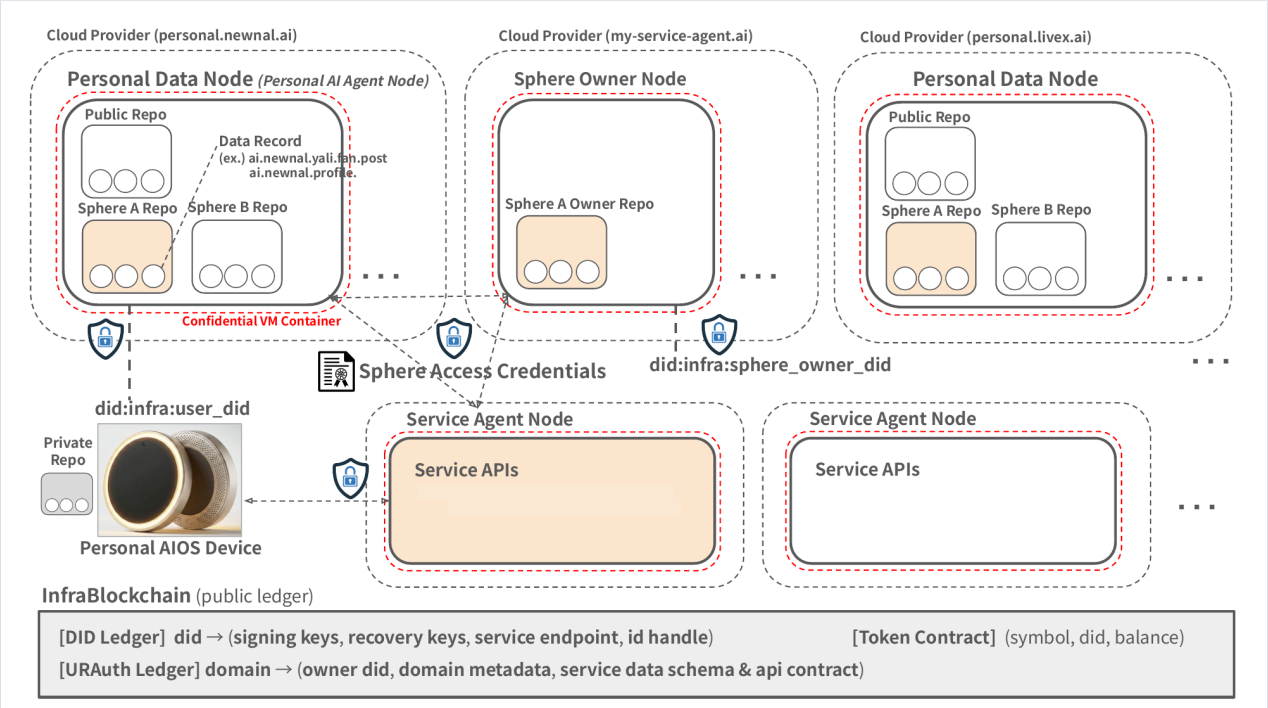
---

02 같은 클라우드, 다른 채움

뉴날은 특별한 저장 공간을 쓰지 않는다. 데이터는 여느 서비스처럼 서버와 클라우드에 저장된다. 다른 것은 **채우는 방식**이다.

- **기존** — 하나의 큰 데이터베이스. 모든 이용자의 식별 정보 · 연락처 · 기록이 계정 키로 묶여 같은 테이블에 쌓인다.

- **뉴날** — 팬 한 사람마다 하나의 **개인 데이터 노드**. 각 노드에는 그 사람의 기록만, 실명이 아니라 탈중앙 신원(DID)에 묶여 담긴다. 어떤 서버에도 두 사람의 데이터가 한 테이블에 담기지 않는다.
- **노드는 물리적 공간이 아니라 논리적 단위다.** 실체는 클라우드 위에 있고, 어느 클라우드든 상관없으며, 언제든 옮길 수 있다. 소유를 만드는 것은 보관 위치가 아니라 — 주소(신원에 묶인다) · 열쇠(기기에만 있다) · 등기(원장에 있다)다.
- **옮길 수 있다는 것이 소유의 실질이다.** 팬은 자기 노드가 어느 클라우드에서 도는지 고를 수 있고, 원장에 등록된 주소만 바꾸면 데이터를 들고 다른 곳으로 간다. 직접 서버를 두는 것도 가능하다. 나갈 수 없는 곳에 맡겨진 데이터는, 얼마나 잘 보호되든 결국 개인의 자산이 아니라 그 회사의 자산이다. 아키텍처 도면에 서로 다른 두 클라우드 주소가 나란히 그려진 이유다.
- **노드 안은 두 칸으로 나뉜다.** 공개해도 되는 기록(포스트 · 댓글 같은 SNS성 데이터)은 **공개 저장소**에, 서비스별로 허락이 필요한 기록은 **스피어 저장소**에 담긴다. 그리고 각 기록에는 미리 공개 등록된 서식이 있다 — 어떤 서비스가 무엇을 담을 수 있는 구조인지가 사전에 정의되어 있고, 누구나 밖에서 읽어볼 수 있다.
- **암호화 키와 기기 밖으로 내보내지 않는 기록은 팬의 기기에만 있다.** 노드는 하드웨어 수준에서 메모리까지 암호화된 **기밀 실행 환경**에서 돌아, 클라우드 운영자도 뉴날도 내용을 볼 수 없다. 관리자 권한이 곧 열람 권한이 되는 통상의 클라우드 운영 모델과 다르다.
- **그리고 그 사실은 밖에서 검증된다.** 기밀 실행 환경은 **원격 증명** 기능을 갖는다 — "공개된 바로 그 코드가 지금 실제로 돌고 있다"를 외부의 제3자가 확인할 수 있다는 뜻이다. 뉴날이 몰래 다른 코드를 올려 두는 경우가 성립하지 않는다.



같은 클라우드 위의 다른 채움 — 팬마다의 개인 데이터 노드(기밀 실행 환경), 기기 안의 프라이빗 저장소, 스피어 소유자와 서비스 에이전트의 노드, 그리고 이 전부를 공증하는 공개 원장(InfraBlockchain).

03 계정 없는 서비스 — 기능별 대체

식별 정보를 저장하지 않아도 서비스가 되는 이유. 계정이 하던 일마다 대체가 있다.

서비스에 필요한 기능	기존의 방법 (식별 정보 필요)	뉴날의 방법 (식별 정보 불필요)
로그인 · 본인 확인	아이디 · 비밀번호, 전화번호 인증	DID와 기기 바인딩 — 기기를 손에 쥔 것이 곧 증명이다
연락 · 알림	전화번호 · 이메일 · 앱 푸시 (공개 주소망)	DID 폐쇄망 통신 — 허락된 상대만 연결된다 (프라이빗 폰 특허)
구독 · 자격 확인	계정에 결제 이력을 묶는다	자격증명 — "유효하다"는 사실만 검증한다
분실 · 복구	주민번호 · 전화번호로 본인 확인	본인 주도의 복구 절차 (미성년 팬은 보호자 참여)
소비자 CS · 환불	계정 조회	판매 채널(파트너 커머스)의 몫 — 오늘도 그 일을 하는 곳
개인화 · 추천	서버의 이용자 프로필에 조인	팬의 기기 안에서, 팬의 키로 조인

연락처가 필요 없는 이유 — 통신망 자체가 다르기 때문이다. 기존의 연락은 공개 주소망 위에서 이루어진다. 전화번호와 이메일은 공개 주소라서, 주소를 아는 사람은 누구나 발신할 수 있다. 스팸과 보이스피싱이 가능한 이유이고, 유출된 연락처가 무기가 되는 이유다. 뉴날의 연락은 DID 기반의 폐쇄망에서 이루어진다 — 전화번호 없이 통화와 메시지가 DID 위에서 작동하는 프라이빗 폰이며, 뉴날의 특허 기술이다. 이 망에는 공개 주소가 없고, 연결은 팬이 허락한 상대와만 성립한다. **도달의 권한이 뒤집힌다 — 공개 주소망에서는 주소를 아는 발신자가 도달을 결정하고, 이 폐쇄망에서는 수신자의 허락이 도달을 결정한다.** DID가 알려져도 달라지는 것은 없다. 허락 없는 발신은 망에서 성립하지 않는다.

약속을 믿는 대신 서식을 읽는다. 계정 시대에는 "우리는 최소한만 받습니다"라는 문장을 믿는 수밖에 없었다. 뉴날에서는 각 서비스가 무엇을 읽고 쓸 수 있는지가 공개 원장에 계약서처럼 등록된다. 그리고 그 서식에는 실명 · 주민번호 · 전화번호 · 계정을 담는 칸이 애초에 없다. 신뢰의 대상이 회사의 문장에서 공개된 설계도로 바뀐다.

계정의 마지막 기능은 **공증**이다 — 누가 누구인지, 무엇이 허용되는지를 회사의 DB가 등기했고, 그래서 회사에 명부가 필요했다. 뉴날 구조에서 이 등기는 공개 원장이 맡는다(8장). 회사가 명부를 가질 이유가 사라진다.

04 개인 데이터 오픈 광장 — 열린 활용, 나뉜 보관

> **개인 데이터 오픈 광장(Personal Data Open Square)은 뉴날 에이전트 플레이스의 중심 컴포넌트다. 여러 회사가 각자 백엔드를 짓지 않고 하나의 개인 데이터 백엔드 위에 서비스를 열고, 팬이 허락한 범위에서만 데이터를 활용하는 열린 공간이다.**

광장이 하는 일은 셋이다.

- **입주** — 파트너는 자신이 다룰 데이터 서식과 API 계약을 공개 원장에 등록하고 서비스를 연다. 회원 데이터베이스도, 신원 관리도, 개인정보 보관도 짓지 않는다.
- **활용** — 원본을 자기 서버로 가져오는 것이 아니라, 팬의 노드에 남은 데이터를 허락받은 범위에서 호출해 쓴다. 보관 책임 없이 활용의 가치만 얻는다.
- **정산** — DID 기준의 장부가 자동 집행한다. 입주사는 서비스 사용료를 받되, 고객의 신원도 원본 데이터도 받지 않는다.

열려 있다는 것은 그 자체로 위험이기도 하다. 서비스가 늘어날수록 한 사람의 여러 도메인 데이터 — 팬 활동, 가족 미디어, 소비, 건강 — 가 같은 개인 데이터 노드에 쌓인다. 그런데 서비스 사이에 칸막이가 없다면, 흩어져 있던 위험이 한 곳에 모이는 셈이 된다. 하나의 서비스가 한 사람의 모든 데이터에 닿을 수 있어서는 안 되고, 자신에게 허락된 것에만 닿아야 한다.

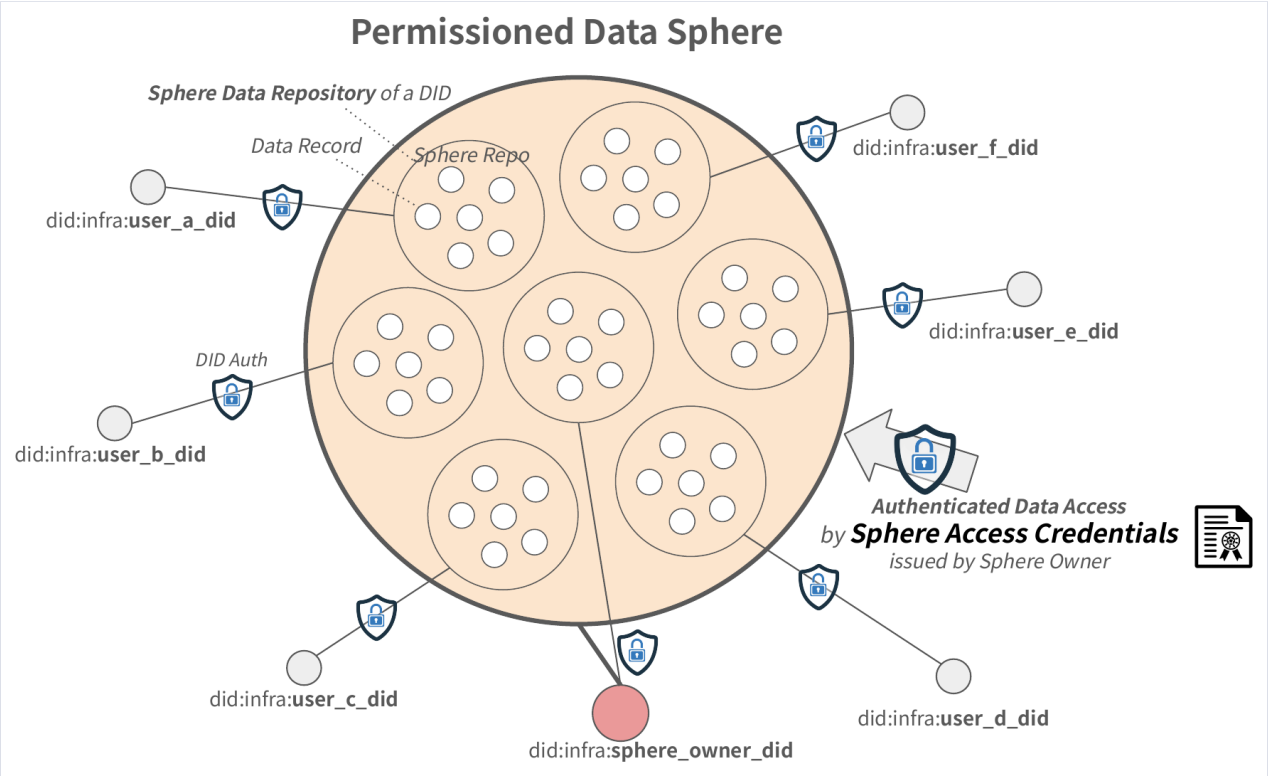
이 문제를 푸는 광장 내부의 구현 기술이 **데이터 스피어**다.

---

05 데이터 스피어 — 광장 안의 칸막이

> **데이터 스피어는 물리적 공간이 아니다. 하나의 서비스 도메인이 여는 데이터 공간이며, 경계는 서비스 도메인(스피어 오너)이 긋고, 그 안을 채우는 것은 팬 각자의 허락이다.**

- **스피어 안은 공용 테이블이 아니다.** 참여자들의 데이터를 한 표에 모아 놓은 것이 아니라, 참여자 DID마다 각각 독립된 저장소가 병렬로 놓인 구조다. 같은 스피어에 있다는 것은 같은 방에 데이터를 넣었다는 뜻이 아니라, 같은 허락의 조건을 공유한다는 뜻이다.
- **데이터는 이동하지 않는다.** 팬이 허락하면 팬의 노드 안에 그 스피어의 칸이 생길 뿐이다. 스피어의 실체는 데이터가 아니라 허락의 목록이다.
- **허락이 바뀌면 경계가 즉시 다시 그려진다.** 철회는 회사의 이행을 기다리는 요청이 아니라, 경계의 재정의다.
- **집단은 명부 없이 만들어진다.** 하나의 스피어에 허락을 그은 팬 전원이 그 스피어의 참여자다 — 전체 DB 없이 "지금 허락 중인 팬들"이라는 집단을 얻고, 집단으로 묶여도 보이는 것은 DID 단위의 기록이지 명부가 아니다.
- **그 전부를 합친 "전체"라는 공간은 시스템 어디에도 없다** — 운영자인 뉴날도 갖고 있지 않다.



데이터 스피어 — 기록(작은 원)은 각자의 노드에 남은 채, 허락이 하나의 가상 공간(큰 원)을 정의한다. 접근은 스피어 소유자가 발급하는 접근 자격증명으로만 이루어지고, 그 한계는 각자가 그은 허락이다.

열쇠는 두 개다

데이터 접근 권한은 **두 겹**으로 되어 있다.

- **스피어 접근 자격증명** — "이 공간에 참여할 자격이 있는가"를 증명한다. 스피어 오너가 발급한다.
- **데이터 주인의 접근 권한 위임** — "내 기록의 이 범위를 열 자격이 있는가"를 증명한다. 팬 본인의 DID 서명이 있어야 한다.

둘 중 하나만으로는 어떤 기록도 열리지 않는다. 서비스 사업자가 스피어의 오너가 되는 경우에도, 오너라는 지위가 참여자 데이터의 열람권을 주지 않는다. 공간을 여는 권한과 내용을 보는 권한이 분리되어 있다.

격리는 세 방향으로 이루어진다

- **오너 대 참여자** — 오너는 공간을 열고 참여 자격을 관리하지만 내용의 보관자가 아니다. 서비스 운영 권한과 데이터 열람 권한이 분리된다.
- **참여자 대 참여자** — 같은 스피어 안에 있어도 다른 참여자의 저장소는 보이지 않는다. 모든 접근은 DID 인증과 그 저장소 주인의 위임을 통과해야 한다.
- **스피어 대 스피어** — 서로 다른 두 스피어는 서로를 조회하는 경로가 없다. **같은 사람의 두 스피어도 마찬가지다.** 서비스가 늘어도 의도치 않은 노출면이 함께 커지지 않는다.

그래서 사용자 입장에서는, 서비스가 늘어날수록 데이터가 회사들에 흩어지지 않고 자기 노드에 쌓이되 **방은 나뉜 채로** 쌓인다. 개인화의 총합은 사용자 쪽에 축적되고, 어느 회사 쪽에도 축적되지 않는다.

---

06 에이전트와 결제 — 두 개의 분리

에이전트 플레이스의 입주사는 **서비스 에이전트 노드**에 자기 서비스를 구현한다. 원본 데이터를 자기 데이터베이스로 끌어오는 대신, 원장에 공표된 API 계약을 통해 팬의 노드를 호출한다. 그 호출은 스피어 오너가 발급한 접근 자격증명과 팬 본인의 위임을 모두 통과해야 성립한다. 받는 것은 서비스의 기회와 정산이고, 받지 않는 것은 신원 · 원본 · 연락처다. **보관의 부담은 뉴날이 지고, 활용은 파트너가 하고, 소유는 팬에게 남는다.**

결제 신원은 오늘도 그 일을 하는 곳(파트너 커머스 · PG)에 남는다. 결제와 서비스는 "이 기기의 구독은 유효하다"는 자격증명으로만 이어진다. 미성년 팬의 결제자는 보호자다 — "미성년자의 결제 정보"가 시스템에 생기지 않는다.

---

07 모든 접근은 기록되고 통보된다

- 허락 없이 일어날 수 있는 활동은 없다 — 읽기 한 번, 검색 한 번까지.
- 모든 접근은 기록되어 데이터의 주인에게 통보된다.
- 소유 · 허락 · 접근 기록은 원장 위에 있어 운영자가 소급 조작할 수 없고, 외부 감사로 검증받는다.
- 기존 구조의 내부자 유출은 권한의 오용이었다. 이 구조에는 팬의 데이터를 열 수 있는 권한 자체가 없다.

---

08 블록체인 — 회사 안에 있던 신뢰의 층을 퍼블릭으로 옮겼다

지금까지 서비스의 신뢰는 운영 회사의 DB 안에 있었다 — 누가 이용자인지, 무엇이 허용되는지, 무슨 일이 있었는지를 회사가 기록했고, 회사만 볼 수 있었다. 이용자에게는 회사를 믿는 것 외의 선택지가 없었다. 뉴날은 이 층을 누구나 확인하고 검증할 수 있는 퍼블릭 원장으로 옮겼다.

원장에 올리는 것은 넷이고, 이를 세 개의 원장이 나눠 맡는다.

- **신원의 등기** — DID, 서명 · 복구 열쇠, 노드의 위치. → **DID 원장**
- **허락의 기록** — 누가 어디까지 열었고, 언제 닫았는가.
- **계약의 공표** — 각 서비스가 읽고 쓸 수 있는 데이터의 범위(데이터 서식과 API 계약), 그리고 그 도메인의 소유자. → **URAuth 원장**
- **정산의 장부** — DID 기준의 자동 정산. → **토큰 컨트랙트**

어느 것도 뉴날의 내부 데이터베이스가 아니고, 원장에 오른 기록은 누구도 임의로 고칠 수 없다.

원본 데이터는 원장에 올리지 않는다. 사연과 기록은 개인의 노드에, 키는 기기에 있다. **블록체인의 역할은 저장이 아니라 공증이다** — 계정, 곧 회사의 등기소가 하던 일을 회사 밖의 원장이 대신한다. 그래서 어느 회사도 명부를 가질 필요가 없다.

이 층의 중심에 데이터의 주인이 있다. 데이터가 어떤 물리적 공간에 있든, 개인의 허락 없이는 어느 회사도 — 뉴날을 포함해 — 데이터를 활용할 수 없고, 그 사실을 누구나 원장에서 확인할 수 있다.

---

09 왜 다른 블록체인으로는 안 되는가

신뢰의 층이 되려면 조건이 있다 — 누구나 검증할 수 있어야 하고(퍼블릭), 비용이 안정적이어야 하고, 실서비스 규모를 감당해야 한다. 기존 퍼블릭 블록체인은 가상화폐 발행과 결합되어 있어 이 조건을 채우지 못한다. 수수료가 코인 시세를 타고, 투기와 규제 리스크가 따라오고, 처리 구조가 실서비스 규모를 감당하지 못한다. 기업과 정부가 쓸 수 없는 물건이었고, 실제로 수천만 명이 매일 쓰는 실생활 서비스를 올린 퍼블릭 체인은 없다.

InfraBlockchain은 가상화폐 발행 없이 작동하는 퍼블릭 블록체인이다. 해당 합의 알고리즘은 미국 · 국제 · 한국 등록 특허다.

---

10 실증 — 4,300만 명의 COOV

백신 패스 COOV는 이 구조가 국가 규모에서 돌아간 실증이다.

- 대한민국 **4,300만 명**이 매일 썼고, **일 최대 3천만 건 이상**의 검증이 돌았으며, **2억 6천만 개 이상의 DID**가 발급되어 사용되었다.
- 다룬 것은 가장 민감한 데이터였다 — 의료 기록, 실명, 주민번호, 여권번호.
- 원본은 발급 기관(질병관리청 · 외교부 등)의 서버에 남았고, 개인의 기기에는 자격증명만 담겼으며, 검증할 때 서버에 묻지 않았다. 어디에도 "털면 전부가 나오는 한 곳"이 없었다.

이 규모의 실서비스 실증을 가진 블록체인 데이터 구조는 세계에 이것뿐이다.

---

11 규제와의 정합

세계 개인정보 규제의 두 잣대는 **식별 가능성**과 **수집이라는 행위**다. 이 구조는 둘 다 갖고 있지 않다.

규제가 요구하는 것	기존 구조의 대응	뉴날 구조
최소 수집 (GDPR 제5조)	수집 항목을 줄이는 노력	식별자 · 연락처를 애초에 수집하지 않는다
동의만큼 쉬운 철회 (제7조)	동의 관리 시스템과 이행 절차	허락과 철회가 같은 동작이다

규제가 요구하는 것	기존 구조의 대응	뉴날 구조
설계 단계의 프라이버시 (제25조)	개발 절차에 심사를 추가	구조 자체가 이 조항의 구현이다
삭제권 (제17조)	흩어진 사본을 찾아 지우고 증명	회사가 사본을 가진 적이 없다
이동권 (제20조)	내보내기 기능을 별도로 구축	데이터가 처음부터 사람에게 묶여 있고, 노드째로 옮긴다
열람권 (제15조)	요청을 접수하고 회신	모든 접근이 기록되고 통보된다
유출 통지 (제33 · 34조)	72시간 대응 체계	"식별된 피해자"가 성립하지 않는 방향
국외 이전 제한 (제5장)	이전 심사와 표준계약	노드 소재지를 관할별로 둔다 — 전체 DB가 국경을 넘는 사건이 없다
아동 보호 (제8조 · 미국 COPPA)	연령 확인과 보호자 동의 절차	아동의 식별자 · 연락처를 수집하지 않으며, 결제와 동의의 주체가 보호자다

미국 주법(캘리포니아 CCPA/CPRA 등)의 핵심 규제 대상 — 개인정보의 판매 · 공유, 데이터 브로커, 무단 표적 광고, 아동 정보 수집 — 은 이 구조에서 행위 자체가 성립하지 않는다. 관할별 세부 적용은 법무 검토와 함께 확정한다.

---

12 왜 따라 하기 어려운가

- **빅테크** — 매출이 데이터 비대칭(표적 광고 · CRM)에서 나온다. 이 구조의 채택은 기능 추가가 아니라 사업 모델의 교체다.
- **기기 회사** — 기기는 만들 수 있어도 계정은 버릴 수 없다. 계정 · 푸시 · 앱스토어 결제가 사업의 뼈대이고, 계정 없는 서비스는 자체 운영체제와 전용 디바이스에서만 성립한다.
- **신뢰의 이력** — 가장 민감한 데이터를 국가 규모로 사고 없이 다룬 이력은 팬데믹이라는 재현 불가능한 조건에서 만들어졌다.
- **스타트업** — 구체 메커니즘은 미국 · 한국 · 유럽 · 국제에 등록 · 출원된 90여 건의 특허로 보호된다.

A 예상 질문과 답 (Q&A)

Q1. 왜 미성년 팬의 데이터를 다뤄도 안전하다고 말하는가? 뉴날의 시스템에는 그 팬이 누구인지 알 수 있는 식별 정보가 없고, 그 팬에게 연락할 수단도 없어서, 유출되어도 쓸 수 없기 때문이다. 유출이 사고가 되는 조건(내용 × 식별 × 도달)에서 식별과 도달이 구조적으로 빠져 있다. 안전의 근거는 보안 등급이 아니라, 훔친 데이터가 범죄의 원료가 되지 못한다는 사실이다.

Q2. 우리도 보안 투자를 많이 한다. 무엇이 다른가? 방향이 다르다. 기존 보안은 뚫리지 않게 막는 일이고, 방어는 언젠가 뚫리며, 데이터를 열 권한을 가진 내부자의 행위는 막지 못한다. 뉴날은 뚫려도 사고가 되지 않고, 열 권한을 가진 내부자가 존재하지 않는 구조다.

Q3. 유저 데이터와 결제 데이터의 분리는 우리도 할 수 있지 않은가? 분리는 누구나 한다. 다른 것은 **조인 키**다. 기존 분리는 두 테이블에 공통 키(계정 · 전화번호)가 남는 정책적 분리라 언제든 다시 붙는다. 뉴날은 조인 키 자체가 시스템에 존재하지 않으며, 내용과 신원이 만나는 유일한 장소는 팬 본인의 기기다. "안 붙이겠다"는 약속과 "붙일 수 없다"는 구조의 차이다.

Q4. 구독과 커머스가 있는데 어떻게 식별 정보가 없다는 것인가? 결제 신원은 존재한다 — 다만 오늘도 그 일을 하는 곳(파트너 커머스 · PG)에 남고, 뉴날은 새로 수집하지 않는다. 결제와 서비스는 자격증명("이 기기의 구독은 유효하다")으로만 이어지며, 광장으로 넘어가는 정보는 그 진위 하나다. 미성년 팬은 결제자가 법정대리인이라 분리가 한 겹 더 깊다.

Q5. 팬이 사연에 자기 이름이나 학교를 쓰면 그것이 식별 정보 아닌가? 팬이 스스로 적는 내용까지 없앨 수는 없다. 그래서 뉴날의 정확한 주장은 "식별이 절대 불가능"이 아니라 **"바깥세상과 잇는 조인 키(법적 신원 · 연락 수단 · 금융 정보)의 구조적 부재"**다. 사연에 이름이 있어도, 그 기록에는 전화번호도 주소도 계정도 붙어 있지 않다 — 범죄에 쓸 연결 고리가 없다.

Q6. 비식별 데이터도 다른 데이터와 결합하면 재식별된다는 연구가 많지 않은가? 그 연구들의 전제는 두 가지다 — 결합할 외부 데이터와 조인할 공통 항목, 그리고 결합의 대상이 되는 전체 데이터셋. 뉴날 구조에는 둘 다 없다. 조인 키가 기록에 붙어 있지 않아 외부 데이터와 이을 고리가 없고, 데이터가 허락 단위의 스피어로만 존재해 "통째로 확보해 결합을 시도할 전체"가 성립하지 않는다. 어떤 기록의 주인을 추정하는 데 성공해도, 그 사람에게 닿는 수단은 여전히 없다.

Q7. 조인이 팬의 기기에서 일어난다면, 팬의 기기를 해킹하면 되지 않는가? 서버 해킹과 기기 해킹은 경제학이 다르다. 서버는 한 번 뚫으면 수천만 명이 나오지만, 기기는 한 대를 뚫어야 한 사람이 나온다 — 그것도 물리적 접근이나 개별 표적 공격을 통해서다. 수고 대비 보상이 성립하지 않아 산업이 되지 못한다. 기기 안의 키와 프라이빗 저장소는 하드웨어 수준에서 보호된다.

Q8. 뉴날 내부자가 열어보면? 뉴날이 해킹당하면? 열 수 없다 — 키가 뉴날에 없기 때문이다. 기존 구조의 내부자 유출은 권한을 가진 사람의 오용이었지만, 이 구조에는 팬의 데이터를 열 수 있는 권한을 가진 사람이 뉴날 안에 없다. 원본은 암호화되어 있고 키는 팬의 기기에 있으며, 노드는 기밀 실행 환경에서 돌아 클라우드 운영자도 내용을 볼 수 없다.

모든 접근은 기록되어 본인에게 통보되고, 이 사실은 외부 감사로 검증받는다. 뉴날이 통째로 해킹당해도 결과는 같다 — 훔쳐 가도 쓸 수 없다.

Q9. 팬이 기기를 잃어버리면 데이터도 잃는가? 아니다. 원본은 암호화되어 분산 보관되고, 키는 본인 주도의 복구 절차(미성년 팬은 보호자 참여)로 되살린다. 복구의 주체는 언제나 본인이며, 뉴날이 대신 열어주는 뒷문은 설계상 존재하지 않는다 — 뒷문은 팬을 위한 것이라도 결국 공격자의 문이 되기 때문이다.

Q10. 수사기관이 데이터를 요구하면? 뉴날이 보유하지 않은 것은 제출할 수 없다 — 뉴날이 낼 수 있는 것은 암호화된 원본과 접근 기록뿐이며, 열람은 데이터의 주인이 응하는 문제다. 이는 종단간 암호화 서비스들이 국제적으로 확립해 온 선례와 같은 구조이며, 바로 그 사실이 팬과 소속사 양쪽을 보호한다.

Q11. 애플이나 구글이 같은 것을 만들면? 만들 수 없는 것이 아니라, 채택할 수 없다. 그들의 사업은 계정과 식별 위에 서 있고 — 광고, CRM, 앱스토어 — 이 구조를 채택하는 순간 자기 사업의 전제를 부정하게 된다. 기술의 문제였다면 이미 나왔을 것이다.

Q12. 개인 데이터 저장소니 데이터 주권이니 하는 시도는 전에도 있지 않았나? 있었다 — 그리고 그 시도들이 멈춘 지점이 뉴날의 출발점이다. 그들은 보관을 개인에게 옮기는 데는 성공했지만 활용을 만들지 못했다. 돌려받은 개인은 데이터로 할 일이 없었고, 기업은 흩어진 데이터를 쓸 방법이 없어 오지 않았다. 뉴날의 구조는 보호와 활용이 하나로 설계되어 있다 — 오픈 광장이 활용의 자리를 만들고, 스피어가 집적 없이 활용 가능한 집단을 만들며, 에이전트가 그 위에서 서비스와 정산을 돌린다.

Q13. 보호가 이렇게 강하면, 정작 우리는 데이터를 활용할 수 있는가? 그것이 이 구조의 목적이다 — 보호는 활용의 반대말이 아니라 조건이다. 파트너는 팬이 허락한 범위를 자연어로 검색하고, 에이전트 서비스를 제안하고, 개인화 콘텐츠를 생성한다. 원본을 가져가는 것이 아니라 허락 위에서 쓰는 것이며, 그래서 보관 책임 없이 가치만 얻는다. 보호가 약하면 활용할 데이터 자체가 들어오지 않는다.

Q14. 법적으로는 어떻게 정리되는가? 뉴날 구조에서는 "수집"이라는 행위 자체가 성립하지 않는 방향이다 — 데이터는 본인 소유로 남고, 회사 자산이 되지 않으며, 처리되는 기록은 회사 관점에서 식별 불가능하다. 조항 단위의 정합은 본문 11장에 있다. 세부는 관할별 법무 검토와 함께 확정한다.

Q15. 그래서, 사고가 나면 누가 책임지는가? 두 겹으로 답한다. 첫째, 이 구조에서는 기존형 유출 사고 — 식별된 개인의 민감 정보가 연락 수단과 함께 새는 사고 — 가 성립하지 않는다. 둘째, 그럼에도 망의 운영과 구조의 유지는 전적으로 뉴날의 책임이며, 그 책임의 이행은 약속이 아니라 감사로 증명한다. 파트너가 지는 데이터 책임은 없다 — 소유하지도, 보관하지도 않기 때문이다.

Q16. 우리가 스피어의 오너라면, 우리는 참여한 팬들의 데이터를 다 볼 수 있는 것 아닌가? 아니다. 오너가 가진 것은 공간을 열고 참여 자격을 관리하는 권한이지, 내용을 여는 권한이 아니다. 어떤 기록이 열리려면 열쇠가 두 개 필요하다 — 오너가 발급한 스피어 접근 자격증명("이 공간에 참여할 자격이 있는가")과, 데이터 주인 본인의 위임("내 기록의 이 범위를 열 자격이 있는가"). 오너는 첫 번째만 갖는다. 운영 권한과 열람 권한이 분리되어 있다는 것이 이 구조의 핵심 중 하나다.

Q17. 서비스가 늘어나면 한 사람의 데이터가 한 노드에 다 모이는 것 아닌가? 그 노드 하나만 뚫으면 되지 않나? 개인 단위로 데이터가 쌓이는 것은 사실이고, 그래서 이 구조는 그 안을 나눈다. 서로 다른 스피어는 서로를 조회하는 경로가 없고, 같은 사람의 두 스피어끼리도 마찬가지다. 하나의 서비스가 허락받은 것은 자기 스피어 안의 정해진 서식뿐이라, 서비스가 늘어도 노출면이 함께 커지지 않는다. 노드 자체는 기밀 실행 환경에서 돌아 클라우드 운영자도 뉴날도 평문을 볼 수 없고, 무엇보다 노드 하나는 한 사람이다 — 수천만 명이 나오는 서버 한 대와 경제학이 다르다.

Q18. 뉴날이 노드의 코드를 몰래 바꿔 놓으면 어떻게 아는가? 그것이 원격 증명의 역할이다. 기밀 실행 환경은 "지금 이 안에서 돌고 있는 코드가 공개된 바로 그 코드"라는 사실을 외부에서 확인할 수 있는 증거를 낸다. 검증하는 쪽은 뉴날이 아니어도 된다. 이 문서의 주장들이 믿음의 대상이 아니라 감사의 대상인 이유다.

> **이 구조는 "뉴날을 믿어라"로 성립하지 않는다. 소유 · 허락 · 접근의 기록은 누구나 검증할 수 있고, 악용에 필요한 정보는 시스템에 존재하지 않으며, 뉴날도 팬의 허락 없이는 데이터를 활용할 수 없다.**

B 기술 구조 요약

본문의 도면에 등장하는 구성 요소를 한 표로 정리한다. 기술 검토를 맡은 분들을 위한 요약이며, 본문의 주장을 이해하는 데 반드시 필요한 내용은 아니다.

구성 요소	하는 일	하지 않는 일
개인 데이터 노드 (Personal Data Node)	개인 데이터의 1차 소재지. 사용자 DID에 귀속되며, 본인 서명이 포함된 접근 허가가 있어야 기록의 생성 · 읽기가 가능하다. 공개 저장소와 스피어 저장소로 나뉘고, 안에서도 암호화되어 분산 저장된다. 클라우드 제공자를 고를 수 있고 언제든 옮길 수 있다.	개인 식별정보를 담지 않는다. 다른 사람의 데이터를 같은 테이블에 담지 않는다.
기밀 실행 컨테이너 (Confidential VM Container)	하드웨어 수준에서 메모리를 암호화 · 봉인한 실행 환경. 원격 증명으로 "공개된 그 코드가 돌고 있음"을 외부에서 검증한다.	클라우드 사업자에게도, 뉴날 운영자에게도 평문을 보여주지 않는다.
개인 AIOS 디바이스	밖으로 내보내지 않는 개인 저장소(Private Repo)와, DID의 최종 소유권을 가진 루트 비밀키를 보관한다. 개인 데이터에 대한 최종 권한은 여기에서 나온다.	DID의 최종 소유권을 가진 루트 비밀키를 외부로 내보내지 않는다.
스피어 오너 노드 (Sphere Owner Node)	서비스 도메인의 데이터 공간, 즉 스피어의 경계와 참여 자격을 정의하고 접근 자격증명을 발급한다.	참여자 데이터를 열람할 권한을 갖지 않는다.
서비스 에이전트 노드 (Service Agent Node)	입주사의 데이터 서비스가 구현되는 곳. 원본을 자기 DB로 끌어오는 대신 공개된 API 계약으로 호출한다.	허락받지 않은 스피어에 접근하지 못한다. 고객 신원과 원본을 받지 않는다.
InfraBlockchain 공개 원장	DID 원장(서명키 · 복구키 · 노드 주소), URAuth 원장(도메인–소유자 DID–데이터 서식 · API 계약), 토큰 컨트랙트(DID 기준 정산 장부).	원본 데이터를 담지 않는다. 뉴날의 내부 DB가 아니며, 누구도 임의로 고칠 수 없다.

하나의 백엔드, 여러 서비스

성격이 다른 서비스들이 같은 개인 데이터 백엔드 위에서, 서로의 데이터에 닿지 않은 채 동작한다.

- **YALI — 팬 개인 데이터 기반 아티스트 AI 기기 서비스.** 팬의 사연 · 감정 · 소비 기록은 팬의 노드 안 YALI 스피어에 남는다. 소속사는 허락된 범위를 검색하고 개인화된 아티스트 콘텐츠를 생성하되, 원본을 자기 서버로 가져가지 않는다.
- **ILLI — 시니어와 가족의 콘텐츠 공유 · Appless AI 에이전트 기기 서비스.** 가족 미디어 공유, 앱을 거치지 않는 상품 추천 · 택시 예약 등. 가족 구성원 여러 사람의 DID가 하나의 스피어를 공유하되 각자의 저장소는 분리된다.

두 서비스가 공유하는 것은 같은 개인 데이터 노드, 같은 DID, 같은 원장이다. 공유하지 않는 것은 데이터다.

입주사 관점 — 백엔드를 짓지 않는 플랫폼

데이터 서식과 API 계약을 URAuth 원장에 등록하고 자신의 스피어를 열면 입주가 끝난다. 회원 데이터베이스를 만들지 않고, 신원을 관리하지 않고, 개인정보를 보관하지 않으므로 **유출 책임의 대상도 되지 않는다.** 정산은 DID 기준의 장부가 자동 집행한다.

사용자 입장에서는 앱도 계정도 로그인도 없다 — 기기와 DID가 그 자리를 대신한다. 관계가 뒤집혀 있어서, 서비스 쪽이 사용자의 노드와 스피어에 로그인한다(Reverse Login).