
AI 시대의 운영체제, 뉴날 aios

개인의 데이터를 퍼스널 데이터 노드 단위로 실행하고 보호하며,
개인이 허락한 범위에서 AI 에이전트를 실행하는 운영체제

운영체제의 정의에서 세 조건을 도출하고,
뉴날 aios가 세 조건을 사람 단위로 충족하는 운영체제임을 설명합니다.

목차

개요	3
왜 새 운영체제가 필요한가	3
aios	3
본 문서의 구성	3
1. 운영체제의 정의	3
2. 기존 운영체제 — 정의의 적용과 한계	4
3. aios의 구성 — 네 구성 요소	5
4. 디바이스가 아닌 사람 단위의 관리 — 퍼스널 데이터 노드	6
5. 인터페이스의 전환	7
5.1 인터페이스로 나뉜 컴퓨팅의 시대	7
5.2 AI 인터페이스가 성립하는 조건	8
5.3 예 — 한 번의 제안이 처리되는 과정	8
6. 정의를 aios에 적용 — 기존 운영체제와의 차이	9
7. 탑재 형태	10
8. 도입 범위 — 단계적 도입	10
9. 예상되는 문제 제기와 답	11
결론	12

개요

왜 새 운영체제가 필요한가

AI 에이전트가 한 사람을 위해 일하려면 세 가지가 필요합니다. 개인의 데이터가 한 곳에 모여 있어야 하고, 개인이 데이터의 사용 범위를 정할 수 있어야 하며, 에이전트가 사람의 명령을 기다리지 않고 데이터를 바탕으로 먼저 제안할 수 있어야 합니다. 기존 운영체제(iOS, 안드로이드, 맥OS, 윈도우)는 세 가지를 디바이스 안의 파일과 앱에 대해서는 수행하지만, 사람의 데이터와 AI 에이전트에 대해서는 수행하지 않습니다. 앱이 데이터를 회사 서버로 가져가는 순간 운영체제의 관리가 끝나고, 이후의 사용 범위는 개인이 아니라 회사가 정하기 때문입니다.

기존 운영체제의 결함이 아닙니다. 기존 운영체제는 디바이스 안의 자원을 관리하고 앱을 실행하는 목적으로 설계되었고, 그 목적은 지금도 수행합니다. AI 시대에 새로 생긴 자원인 개인의 데이터와 새로 생긴 실행 대상인 AI 에이전트를 사람 단위로 관리하는 운영체제가 필요해졌을 뿐입니다.

aios

뉴날 aios는 개인의 데이터를 퍼스널 데이터 노드 단위로 실행하고 보호하며, 퍼스널 데이터 노드 환경에서 AI 에이전트를 실행하는 운영체제입니다. aios는 디바이스가 있어도 없어도, 기존 운영체제 위에서도 기존 운영체제 없이도 동작합니다. 전용 디바이스에서는 하드웨어까지 관리하는 독립 운영체제로, 기존 운영체제가 탑재된 디바이스에서는 기존 운영체제 위의 운영체제 계층으로, 디바이스가 없는 서비스 안에서는 노드를 실행하는 운영체제 계층으로 동작하며, 세 경우 모두에서 수행하는 일은 같습니다.

운영체제가 자원을 다루려면 자원에 관리 단위가 있어야 합니다. 기존 운영체제는 저장을 파일 단위로, 실행을 프로세스 단위로 관리합니다. aios는 개인의 데이터와 데이터에 대한 실행을 퍼스널 데이터 노드 단위로 관리합니다. 노드는 개인 한 사람마다 하나씩 실행되며 데이터를 보관하고 허락을 집행하고 서비스를 실행하는 프로그램입니다. 기존 운영체제가 프로세스를 생성·격리·보호하듯, aios는 노드를 생성하고(개인마다 하나), 격리하고, 보호하고, 이동시킵니다. 노드가 있기 때문에 aios는 사람 단위로 자원을 관리하고, 접근을 통제하고, 에이전트를 실행합니다. aios에서는 개인의 데이터뿐 아니라 서비스가 이를 활용해 새롭게 생산한 데이터도 개인의 노드에 축적되며, 개인의 허락에 따라 다른 서비스에서 다시 활용될 수 있습니다. 기업은 aios의 일부(데이터 계층)부터 전체까지 범위를 정해 도입합니다.

본 문서의 구성

1장은 운영체제의 교과서 정의에서 세 조건을 도출합니다. 2장은 기존 운영체제가 세 조건을 어느 단위로 수행하는지와 AI 에이전트에 필요한 단위를 비교합니다. 3장은 aios의 네 구성 요소를, 4장은 사람 단위의 관리와 퍼스널 데이터 노드를, 5장은 인터페이스의 전환과 한 번의 제안이 처리되는 예를, 6장은 정의를 aios에 적용한 결과와 기존 운영체제와의 차이를, 7장은 탑재 형태, 8장은 단계적 도입 범위, 9장은 예상되는 문제 제기와 답을 다룹니다.

1. 운영체제의 정의

운영체제 교과서의 정의는 다음과 같습니다.

- Tanenbaum, Modern Operating Systems: 운영체제는 두 가지 역할을 한다. 하드웨어의 복잡한 세부를 감추고 프로그램에 깨끗한 추상(프로세스, 파일 등)을 제공하는 확장된 기계(extended machine)의 역할과, 프로세서·메모리·저장장치·입출력 장치를 여러 프로그램이 나누어 쓰도록 배분하고 보호하는 자원 관리자(resource manager)의 역할이다.
- Silberschatz, Operating System Concepts: 운영체제는 컴퓨터 하드웨어를 관리하고, 응용 프로그램의 기반을 제공하며, 사용자와 하드웨어 사이의 중개자 역할을 하는 프로그램이다.

두 정의에 공통으로 포함된 운영체제의 조건은 세 가지입니다.

조건	내용
자원 관리	운영체제는 자원을 여러 프로그램이 나누어 쓰도록 배분하고 보호합니다. 자원에는 프로세서·메모리·장치와 함께 파일과 데이터가 포함됩니다.
접근 통제	운영체제는 어떤 프로그램과 사용자가 어떤 자원에 접근할 수 있는지를 정하고 강제합니다.
실행 기반	운영체제는 응용 프로그램이 동작할 수 있는 추상과 인터페이스를 제공하고, 프로그램을 실행합니다.

정의에 포함되지 않은 것도 명시합니다. 정의는 운영체제가 자체 커널로 디바이스를 부팅할 것을 요구하지 않고, 관리하는 자원을 하드웨어에 한정하지 않으며, 다른 소프트웨어 위에서 동작하는 것을 배제하지 않습니다. 안드로이드는 리눅스 커널 위에서 동작하지만 운영체제로 불립니다. 판단 기준은 세 조건을 수행하는지 여부입니다.

정의가 요구하는 것이 하나 더 있습니다. 자원은 운영체제가 다룰 수 있는 **관리 단위**를 가져야 합니다. 기존 운영체제는 저장 자원을 파일 단위로, 실행을 프로세스 단위로 관리합니다. 파일과 프로세스라는 단위가 있기 때문에 운영체제는 자원을 배분하고, 접근을 통제하고, 프로그램을 실행할 수 있습니다. AI 시대의 운영체제가 개인의 데이터를 자원으로 관리하려면 개인의 데이터에도 관리 단위가 있어야 합니다. 뉴날 aios에서 관리 단위가 퍼스널 데이터 노드이며, 노드는 파일처럼 데이터를 보관하고 프로세스처럼 실행되는 단위입니다.

2. 기존 운영체제 — 정의의 적용과 한계

기존 운영체제 가운데 iOS와 안드로이드를 예로 들면 다음과 같습니다. macOS와 윈도우도 같은 구조입니다.

조건	iOS	안드로이드
자원 관리	애플 디바이스의 하드웨어와 디바이스 안의 파일	여러 제조사 디바이스의 하드웨어와 디바이스 안의 파일
접근 통제	앱 샌드박스, 카메라·연락처 등 디바이스 자원에 대한 사용자의 권한 허락	앱 샌드박스, 디바이스 자원에 대한 사용자의 권한 허락
실행 기반	앱 실행	앱 실행

기존 운영체제의 접근 통제는 디바이스 안의 자원까지만 미칩니다. 앱이 이용자의 데이터를 앱 회사의 서버로 가져가면, 서버의 데이터는 운영체제의 관리와 통제 밖에 있습니다. 같은 사람의 데이터가 앱마다 흩어져 각 회사의 서버에 보관되고, 어느 운영체제도 서버의 데이터를 자원으로 관리하지 않습니다. 기존 운영체제에는 개인의 데이터를 다루는 단위가 없기 때문입니다. 파일은 디바이스 안에서만 단위이고, 서버로 간 데이터는 회사의 데이터베이스 행이 됩니다. 서비스가 개인의 데이터를 분석·가공하여 새롭게 생산한 데이터 역시 해당 서비스에 머무르며, 다른 서비스에서 재사용하기 어렵습니다.

개요에서 말한 세 가지 — 데이터가 한 곳에, 사용 범위는 개인이, 에이전트는 먼저 — 를 운영체제의 세 조건에 대응시키면 다음과 같습니다. 기존 운영체제는 세 조건 모두를 디바이스와 앱 단위로 수행하도록 설계되었고, 사람 단위로는 수행하지 않습니다.

조건	기존 운영체제가 수행하는 단위	AI 에이전트에 필요한 단위	기존 운영체제에 없는 것
자원 관리	디바이스 안의 하드웨어와 파일	한 사람의 데이터 전체	한 사람의 데이터가 앱마다 흩어져 각 회사의 서버에 있고, 운영체제는 서버의 데이터를 관리하지 않습니다.
접근 통제	디바이스 자원에 대한 앱의 권한	사람의 데이터에 대한 에이전트의 권한	데이터가 디바이스를 벗어나 회사 서버로 가면 운영체제의 통제가 끝납니다. 사용 범위는 개인이 아니라 회사가 정하고, 허락과 접근의 기록도 회사 안에만 남습니다.
실행 기반	이용자의 명령을 기다리는 앱	데이터를 바탕으로 먼저 제안하는 에이전트	앱은 자기 회사 서버의 데이터만 보고, 다른 회사 서버에 있는 같은 사람의 데이터에는 접근할 수 없습니다.

기존 운영체제 위에서 AI 에이전트를 만드는 방법은 세 가지입니다. 한 앱의 데이터만 보는 에이전트를 만들거나, 사람이 필요할 때마다 데이터를 직접 전달하거나, 여러 앱의 데이터를 한 회사의 서버에 모으는 것입니다. 첫째 방법은 사람 전체를 알지 못하므로 맞춤 서비스가 되지 않고, 둘째 방법은 사람이 명령해야 동작하므로 먼저 제안하지 못하며, 셋째 방법은 한 회사에 개인 데이터가 집중되어 유출과 오용의 위험이 커집니다. 어느 방법도 운영체제가 사람 단위의 자원 관리와 접근 통제를 담당하지 않는 한 해결되지 않습니다.

3. aios의 구성 — 네 구성 요소

운영체제의 세 조건에 따라 aios의 네 구성 요소는 다음과 같이 나뉩니다. 네 구성 요소는 탑재 형태(7장)와 관계없이 같고, 클라우드와 AI 모델 같은 기존 인프라를 대체하지 않고 부품으로 사용합니다.



그림 1. aios의 네 구성 요소. 위는 탑재 형태(7장), 아래는 aios가 부품으로 사용하는 기존 인프라입니다.

조건	구성 요소	하는 일
자원 관리	퍼스널 데이터 노드	개인마다 하나씩 실행되는 노드가 개인의 데이터를 개인의 이름으로 보관하고, aios가 노드를 생성·격리·보호·이동시킵니다.
접근 통제	인프라블록체인	소유·허락·접근의 모든 기록을 기업 밖의 공개 원장에서 검증합니다. 가상화폐 발행 없이 동작하는 퍼블릭 원장이라 기업과 정부가 사용할 수 있고, 뉴날도 기록을 바꿀 수 없습니다.
실행 기반	AI 인터페이스	노드의 데이터를 바탕으로 AI가 먼저 제안하고, 사람은 승인만 합니다.
실행 기반	에이전트 플레이스	검증된 에이전트들이 개인이 허락한 노드의 데이터 위에서 서비스를 확장합니다.

네 구성 요소 가운데 설명이 더 필요한 둘을 설명합니다. **에이전트 플레이스**에 등록되는 에이전트는 두 조건을 충족합니다. 읽고 쓸 데이터의 범위가 공개 원장에 등록되어 있고, 실행 코드가 공개된 코드와 같다는 사실을 원격 증명으로 증명할 수 있습니다. 두 조건을 충족한 에이전트를 "검증된 에이전트"라고 부르며, 파트너 기업이 만든 에이전트와 외부 제작자의 에이전트가 같은 조건으로 등록됩니다. 사용 기록은 DID 단위 장부에 기록되어 자동 정산됩니다. **AI 인터페이스**는 제안과 승인의 반복입니다. AI가 노드의 데이터로 제안하고, 사람이 승인하거나 거절하며, 승인은 공개 원장에 허락 기록으로 남습니다. 사람이 명령하지 않아도 동작하지만, 사람의 승인 없이는 어떤 동작도 실행되지 않습니다.

4. 디바이스가 아닌 사람 단위의 관리 — 퍼스널 데이터 노드

1장의 세 조건을 사람 단위로 수행하려면 "한 사람의 데이터 전체와 데이터에 대한 실행"이 운영체제가 다룰 수 있는 하나의 단위여야 합니다. 뉴날 aios는 관리 단위를 **퍼스널 데이터 노드(Personal Data Node)**로 정합니다.

퍼스널 데이터 노드는 개인 한 사람마다 하나씩 실행되는 프로그램입니다. 노드는 개인의 데이터를 암호화해 보관하고, 개인이 허락한 범위를 집행하며, 서비스와 AI 에이전트를 퍼스널 데이터 노드 환경에서 실행합니다.

노드는 클라우드의 기밀 컴퓨팅 환경에서 실행되고, 개인의 DID로 공개 원장에 등록되어 있으며, 개인의 디바이스에 있는 암호화 키로만 열립니다. 개인은 노드를 다른 클라우드로 옮길 수 있습니다. 노드에는 개인의 데이터가 출처를 가리지 않고 개인의 이름으로 모입니다 — 디바이스가 만든 데이터, 기관이 발급한 자격증명, 서비스가 개인의 허락 아래 기록한 데이터가 모두 한 노드에 있습니다.

퍼스널 데이터 노드에 보관되는 개인 데이터에는 발급 주체(Issuer)의 디지털 서명이 포함될 수 있으며, 이를 통해 데이터의 발급 주체와 무결성을 검증할 수 있습니다.

서비스는 개인이 허락한 데이터를 활용하여 새로운 분석·가공·개인화 데이터를 생산하고, 자신의 디지털 서명을 담아 개인의 노드에 반환할 수 있습니다. 이렇게 축적된 데이터는 개인의 허락에 따라 다른 서비스에서 다시 활용될 수 있습니다.

노드가 운영체제의 세 조건에 어떻게 대응하는지는 다음과 같습니다.

조건	퍼스널 데이터 노드의 역할
자원 관리	aios가 관리하는 자원은 개인의 데이터이고, 관리 단위는 노드입니다. aios는 노드를 생성하고(개인마다 하나), 격리하고(노드 간 분리), 보호하고(암호화와 기밀 실행), 이동시킵니다(클라우드 간 이동). 기존 운영체제가 파일과 프로세스에 대해 수행하는 관리를 aios는 노드에 대해 수행합니다.
접근 통제	접근 통제를 노드가 집행합니다. 서비스나 에이전트가 노드에 접근을 요청하면 개인이 허락하고, 허락 범위 밖의 호출은 노드가 실행하지 않습니다. 허락과 접근의 기록은 노드 밖의 공개 원장에 남습니다.
실행 기반	AI 에이전트를 노드가 실행합니다. 에이전트는 퍼스널 데이터 노드 환경에서, 개인이 허락한 범위의 데이터로 개인화된 결과를 만들며, 데이터를 외부로 전송하지 않고 처리할 수 있습니다.

노드가 없으면 aios는 성립하지 않습니다. 관리할 자원의 단위가 없고, 허락을 집행할 주체가 없으며, 에이전트를 실행할 기반이 없기 때문입니다. 반대로 노드가 있으면 세 조건이 노드 하나에서 수행됩니다. 노드가 aios의 관리 단위인 이유는 설계의 선택이 아니라 정의의 요구입니다.

노드와 디바이스의 관계는 다음과 같습니다. 노드는 클라우드에서 실행됩니다. 노드를 여는 키는 개인의 디바이스에만 있습니다. aios는 디바이스에 탑재되든 서비스 안에 탑재되든 개인의 디바이스에 있는 키로 노드를 열어 에이전트를 실행합니다. 디바이스가 전용 디바이스인지, 기존 운영체제가 탑재된 스마트폰인지는 aios의 동작 방식을 바꾸지 않습니다. 디바이스의 전원이 꺼져 있어도 노드는 동작하고, 디바이스를 분실해도 노드의 데이터는 개인이 정한 복구 수단으로 복구됩니다. 노드의 구조와 키 관리, 보안은 별도 문서 「퍼스널 데이터 노드」에서 설명합니다.

5. 인터페이스의 전환

5.1 인터페이스로 나뉜 컴퓨팅의 시대

컴퓨팅의 시대는 운영체제의 인터페이스로 나뉘었습니다. 소비자는 커널이나 자원 관리 방식이 아니라 인터페이스로 새 컴퓨팅을 경험했습니다.

시대	운영체제	인터페이스	동작 방식
퍼스널 컴퓨팅	맥OS, 윈도우	그래픽 인터페이스	사람이 마우스와 키보드로 명령합니다.
모바일	iOS, 안드로이드	멀티터치	사람이 앱을 골라 손가락으로 명령합니다.
AI	aios	AI 인터페이스	AI가 노드의 데이터를 바탕으로 먼저 제안하고, 사람은 승인합니다.

앞의 두 시대의 인터페이스는 모두 사용자 인터페이스입니다. 사람이 명령하고 기계가 수행합니다. aios의 인터페이스는 방향이 반대입니다. AI가 개인의 데이터를 바탕으로 먼저 제안하고, 사람은 승인만 합니다.

5.2 AI 인터페이스가 성립하는 조건

AI가 먼저 제안하려면 AI가 개인의 데이터 전체를 알아야 하고, 접근을 개인이 허락해야 합니다. 두 조건은 노드가 있을 때만 성립합니다. 데이터 전체를 한 노드가 보관해야 AI가 전체를 알 수 있고, 접근 통제가 노드에서 실행되어야 사람이 허락할 수 있습니다. 기존 운영체제의 음성 비서가 디바이스 안의 데이터와 자기 회사 서비스 안에서만 동작하는 이유는 두 조건이 없기 때문입니다. AI 인터페이스는 aios가 개인의 데이터를 노드 단위로 관리하기 때문에 가능한 결과이며, 사용자가 aios를 새 운영체제로 경험하는 지점입니다.

5.3 예 — 한 번의 제안이 처리되는 과정

세 조건과 인터페이스가 실제로 어떻게 결합되는지를 한 번의 제안이 처리되는 과정으로 제시합니다. 예는 시니어 가정의 가족 전용 전화이지만, 은행 앱이나 병원 서비스 안의 aios도 같은 순서로 동작합니다.

순서	일어나는 일	운영체제의 조건
1	디바이스가 전날 밤부터 움직임이 없다는 생활 데이터를 개인의 노드에 기록합니다. 병원이 발급한 검진 결과(자격증명)도 같은 노드에 있습니다.	자원 관리 — 데이터가 개인의 노드에 모입니다.
2	가족 돌봄 서비스의 에이전트가 노드에 접근을 요청합니다. 요청할 수 있는 범위는 공개 원장에 등록된 범위(움직임·안부 응답)뿐이고, 검진 결과는 범위 밖이므로 요청 자체가 실행되지 않습니다.	접근 통제 — 허락이 노드에서 집행됩니다.
3	에이전트가 퍼스널 데이터 노드 환경에서 개인이 허락한 데이터로 판단하고, "오늘 안부 확인 전화를 걸까요?"라고 먼저 제안합니다. 데이터는 퍼스널 데이터 노드 환경 밖으로 전송되지 않았습니다.	실행 기반 — 에이전트가 노드에서 실행됩니다.
4	개인(또는 보호자)이 디바이스에서 승인합니다. 승인과 접근이 공개 원장에 기록되고 통보됩니다.	AI 인터페이스 — AI가 제안하고 사람이 승인합니다.
5	통화가 연결됩니다. 연결은 전화번호가 아니라 DID 기반 폐쇄망으로 이루어지므로 등록되지 않은 상대는 연결 경로가 없습니다.	접근 통제 — 연락도 허락의 대상입니다.

순서	일어나는 일	운영체제의 조건
6	서비스 사용 기록이 DID 단위 장부에 기록되어 사용료가 정산됩니다. 가족 돌봄 서비스 회사는 개인의 데이터를 보관한 적이 없습니다.	자원 관리 — 정산도 노드 단위입니다.

6. 정의를 aios에 적용 — 기존 운영체제와의 차이

조건	aios
자원 관리	개인의 데이터를 퍼스널 데이터 노드 단위로 관리합니다. 노드는 개인마다 하나씩 실행되고, 개인의 DID로 공개 원장에 등록되며, 개인의 키로만 열립니다. 전용 디바이스에서는 디바이스의 하드웨어도 관리합니다.
접근 통제	DID로 식별된 개인의 허락. AI 에이전트는 허락받은 범위의 데이터만 노드에서 호출해 사용하고, 허락과 접근의 기록은 공개 원장(인프라블록체인)에 남습니다.
실행 기반	AI 에이전트 실행. AI가 노드의 데이터를 바탕으로 먼저 제안하는 AI 인터페이스와, 검증된 에이전트가 배포·실행되는 에이전트 플레이스를 제공합니다.

aios는 세 조건을 모두 수행합니다. 관리하는 자원이 개인의 데이터이고 관리 단위가 노드라는 점, 접근 통제가 디바이스 안이 아니라 노드에서 실행된다는 점, 실행 대상이 앱이 아니라 AI 에이전트라는 점이 기존 운영체제와 다르고, 운영체제라는 사실 자체는 정의에 따라 성립합니다.

	기존 운영체제	aios
관리하는 자원	디바이스의 하드웨어와 디바이스 안의 파일	개인의 데이터 (전용 디바이스에서는 하드웨어 포함)
관리 단위	파일, 프로세스	퍼스널 데이터 노드 (한 사람에 하나, 데이터를 보관하고 실행되는 프로그램)
접근 통제가 실행되는 곳	디바이스 안	노드. 서비스와 에이전트의 모든 접근 요청을 노드가 집행
접근을 허락하는 주체	디바이스 사용자	DID로 식별된 개인
허락 기록의 위치	디바이스 안	기업 밖의 공개 원장
실행 대상	앱	AI 에이전트
인터페이스	사람이 명령하는 사용자 인터페이스	AI가 먼저 제안하는 AI 인터페이스

기존 운영체제에서는 이용자가 앱에 로그인하고 앱이 이용자의 데이터를 가져가며, 가져간 뒤의 사용 범위는 앱을 만든 회사가 정합니다. aios에서는 서비스와 AI 에이전트가 개인의 노드에 접근을 요청하고, 개인이 허락한

범위에서 데이터를 활용합니다. 뉴날은 이러한 접근 관계의 전환을 역로그인(Reverse Login)이라고 부릅니다.

정의에 따른 한 문장은 다음과 같습니다. aios는 개인의 데이터를 퍼스널 데이터 노드 단위로 실행하고 보호하며, 개인의 허락을 기준으로 노드의 접근을 통제하고, 퍼스널 데이터 노드 환경에서 AI 에이전트를 실행하는 운영체제입니다.

7. 탑재 형태

aios는 세 가지 형태로 탑재됩니다. 어느 형태든 노드를 실행하고 보호하고, 접근을 통제하고, AI 에이전트를 실행하는 방식은 같습니다.

뉴날 aios	전용 디바이스	기존 OS가 탑재된 디바이스	서비스 안의 aios (디바이스 없음)
하드웨어 관리	aios가 수행	기존 OS가 수행	해당 없음
퍼스널 데이터 노드 관리	aios가 수행	aios가 수행	aios가 수행
접근 통제	aios가 수행	aios가 수행	aios가 수행
AI 인터페이스	디바이스 화면에서	앱 안에서	파트너의 앱·웹·상담 채널 안에서
AI 에이전트 실행	aios가 수행	aios가 수행	aios가 수행 (퍼스널 데이터 노드 환경)
예	가족 전용 전화	안드로이드 TV, 스마트워치	은행 앱, AI 콘택트센터, 병원 서비스

전용 디바이스에서 aios는 하드웨어 관리까지 직접 수행하는 독립된 운영체제로 동작합니다. 기존 OS가 탑재된 디바이스에서는 하드웨어 관리를 기존 OS에 위임하고 기존 OS 위에서 퍼스널 데이터 노드의 보안 계층으로 동작합니다. 디바이스가 없는 서비스에서는 클라우드의 노드 런타임과 파트너 서비스 안의 aios 계층으로 동작합니다. 세 형태 모두 운영체제의 세 조건을 수행합니다.

8. 도입 범위 — 단계적 도입

기업은 aios의 네 구성 요소를 한 번에 도입하지 않아도 됩니다. 데이터 계층부터 도입하고 단계적으로 범위를 확장할 수 있으며, 어느 범위에서든 도입하는 대상은 aios입니다. 퍼스널 데이터 노드와 인프라블록체인은 항상 함께 도입됩니다. 노드의 등록과 허락 기록이 공개 원장에 있기 때문입니다. 따라서 최소 도입 단위는 노드가 아니라 aios의 데이터 계층입니다.

도입 범위	포함되는 구성 요소	기업의 이점
데이터 계층	퍼스널 데이터 노드 + 인프라블록체인	개인의 허락에 따른 데이터 활용, 데이터 스피어의 합, 발급 주체와 무결성 검증, 서비스가 생산한 데이터의 축적과 재사용
+ AI 인터페이스	위 구성 + AI 인터페이스	기업의 앱·채널 안에서 AI가 노드의 데이터로 먼저 제안하고, 고객은 승인만 합니다.
+ 에이전트 플레이스	위 구성 + 에이전트 플레이스	기업이 만들지 않은 기능을 검증된 외부 에이전트가 허락된 데이터 위에서 수행합니다.
전용 디바이스	위 구성 전부 + 하드웨어 관리	처음부터 aios를 위해 설계된 디바이스 경험

어느 범위로 도입하든 고객마다 퍼스널 데이터 노드가 생성됩니다. 뉴날이 공급하는 것은 aios이며, 뉴날의 핵심 지표는 활성 노드 수입입니다.

9. 예상되는 문제 제기와 답

커널을 갖고 디바이스를 부팅하지 않으면 운영체제가 아니다. 정의는 커널 소유를 조건으로 두지 않습니다. 안드로이드는 리눅스 커널 위에서 동작하는 운영체제입니다. aios는 전용 디바이스에서는 커널을 포함해 디바이스를 부팅하고, 기존 OS가 탑재된 디바이스에서는 하드웨어 관리를 기존 OS에 위임합니다. 어느 쪽이든 세 조건의 수행은 같습니다.

기존 OS 위에서 동작하면 앱이다. 앱과 운영체제를 가르는 기준은 동작하는 위치가 아니라 역할입니다. 앱은 자원 관리·접근 통제·실행 기반을 운영체제로부터 제공받는 쪽이고, 운영체제는 제공하는 쪽입니다. aios는 개인의 데이터를 퍼스널 데이터 노드 단위로 관리하고, 노드에 대한 접근을 통제하며, 다른 프로그램(AI 에이전트)에 실행 기반을 제공합니다. 세 가지 모두 앱이 하지 않는 일입니다.

개인의 데이터는 하드웨어가 아니므로 운영체제의 자원이 아니다. 정의가 말하는 자원은 하드웨어에 한정되지 않습니다. 파일 시스템은 모든 운영체제의 핵심 구성 요소이고, 파일과 데이터는 운영체제가 관리하는 자원입니다. aios가 관리하는 퍼스널 데이터 노드는 데이터와 데이터에 대한 실행을 함께 가진 자원이며, 기존 운영체제에서 파일과 프로세스가 차지하는 자리를 사람 단위로 차지합니다.

퍼스널 데이터 노드는 데이터베이스나 클라우드 저장소의 다른 이름이다. 데이터베이스와 클라우드 저장소는 데이터를 저장하고 조회하지만, 다른 프로그램에 실행 기반을 제공하지 않고, 접근 통제의 기준이 저장소 소유 회사의 계정입니다. 퍼스널 데이터 노드는 접근 통제의 기준을 개인의 DID와 허락에 두고, 허락 범위 밖의 호출을 실행하지 않으며, 에이전트가 퍼스널 데이터 노드 환경에서 실행됩니다. 노드가 실행되는 클라우드와 노드의 데이터가 놓이는 저장 공간은 aios 안에서 쓰이는 부품이지 노드의 대체물이 아닙니다.

AI 에이전트는 앱의 다른 이름이다. 앱은 이용자의 명령을 받아 회사 서버의 데이터로 동작합니다. AI 에이전트는 개인이 허락한 범위의 데이터를 노드에서 호출해 먼저 제안하고, 허락 범위 밖의 데이터는 사용할 수 없습니다. 실행

대상의 동작 방식과 접근 방식이 다르므로, 실행 기반도 달라야 합니다. aios가 제공하는 실행 기반은 에이전트의 접근 요청과 개인의 허락, 허락의 기록을 처리하는 인터페이스이며, 기존 운영체제에는 없습니다.

기존 운영체제의 음성 비서도 AI 인터페이스다. 기존 운영체제의 음성 비서는 디바이스 안의 데이터와 자기 회사 서비스 안에서 사람의 명령을 받아 수행합니다. 사람이 명령하는 사용자 인터페이스의 한 형태이고, 그 범위는 운영체제가 관리하는 자원의 범위와 같습니다. 개인의 데이터 전체를 바탕으로 먼저 제안하는 인터페이스는 운영체제가 개인의 데이터를 노드 단위로 관리해야 가능하고, aios의 AI 인터페이스는 노드의 데이터로 동작합니다.

디바이스를 구동하지 않으면 운영체제가 아니다. 정의의 세 조건에 하드웨어 구동은 없습니다. 선례가 있습니다. ROS(Robot Operating System)는 리눅스 위에서 동작하며 커널도 하드웨어 구동도 직접 하지 않지만, 로봇의 자원을 추상화하고 구성 요소 간 통신과 실행 기반을 제공하므로 전 세계 로봇 업계가 운영체제로 분류합니다. aios는 디바이스가 없는 서비스 안에서도 퍼스널 데이터 노드를 실행하고 보호하고, 노드의 접근을 통제하고, AI 에이전트를 실행합니다. 디바이스는 aios가 하드웨어 관리까지 수행하는 경우이지 운영체제의 조건이 아닙니다.

운영체제라는 말은 마케팅 표현이다. 판단 기준은 표현이 아니라 정의입니다. 6장의 표가 세 조건의 수행 여부를 항목별로 보여 줍니다. 세 조건 중 하나라도 수행하지 않는다는 반론이 있으면 그 항목을 기준으로 논의하면 됩니다.

결론

운영체제의 정의가 요구하는 조건은 자원 관리, 접근 통제, 실행 기반의 세 가지이며, 하드웨어 구동은 조건에 포함되지 않습니다. 기존 운영체제는 세 조건을 디바이스의 파일과 프로세스, 그리고 앱에 대해 수행합니다. 뉴날 aios는 같은 세 조건을 사람 단위로 수행합니다. 개인의 데이터를 퍼스널 데이터 노드 단위로 실행하고 보호하며, 개인의 허락을 기준으로 노드의 접근을 통제하고, AI가 먼저 제안하는 인터페이스로 퍼스널 데이터 노드 환경에서 AI 에이전트를 실행합니다. 디바이스가 있어도 없어도, 기존 운영체제 위에서도 기존 운영체제 없이도 동작하며, 어느 형태로 도입하든 고객마다 퍼스널 데이터 노드가 생성됩니다. aios에서는 발급 주체의 디지털 서명이 포함된 데이터의 출처와 무결성을 검증할 수 있으며, 서비스가 새롭게 생산한 데이터 역시 개인의 노드에 축적되어 다른 서비스에서 재사용될 수 있습니다.

참고 문헌 Andrew S. Tanenbaum, Herbert Bos, Modern Operating Systems, 4th ed., Pearson, 2015, Chapter 1. / Abraham Silberschatz, Peter B. Galvin, Greg Gagne, Operating System Concepts, 10th ed., Wiley, 2018, Chapter 1.