

출처: Hitachi 주간 사보 5월호

---

## 5 컴포넌트기술과 관리기구

### 5.1 Component Base Technology

Component Base Platform은, 프랑스의 넥스웨이브 솔루션 사(이하, “넥스웨이브”라고 함.)가 개발한 “NSI(NexWave Software Infrastructure)”의 핵심 기술로, 넥스웨이브와 하다찌그룹이 공동으로 개발을 진행하고 있는 컴포넌트 관리기구에 의하여 실현되었다. NSI(NexWave Software Infrastructure)는, 넥스웨이브가 개발한 BUILDER시스템(build & compile system)을 중심으로 하여 컴포넌트 지향 개발에 필요로 하고 기반이 되는 모든 요소와 환경을 포함하고 있다. 특정의 OS에 의존하지 않고, 모듈의 호출 관계에만 초안해서 소프트웨어의 컴포넌트화를 실현한다. NSI를 적용한 BUILDER시스템(build & compile system)에서는, 데이터와 프로그램을 캡슐화하여, 컴포넌트로서 관리하고, 사전에 규정한 인터페이스에 의거해, 컴포넌트 사이의 연결 관계를 동적으로 연결 처리한다. 또한 규정한 인터페이스를 입력, 선언, 개입시키지 않으면 컴포넌트를 access할 수 없기 때문에, 시스템 별로 호출 관계의 인터페이스의 정합성을 자동 확인 연결하고 컴포넌트의 재이용이 가능한 환경을 제공 한다.

NSI는 여러 종류의 MPU에 대한 처리 환경을 제공 할 수 있기 때문에, 컴포넌트는 하나의 MPU상에 복속 되어 존재할 필요가 없다. NSI는, 종래의 BUILDER(build & compile system) 시스템에서는 실현이 곤란하다고 여겨졌던 우수한 기능을 가진다. (표1 참조). 한걸음 더 나가, direct binding 기능을 적용시켜, 컴포넌트 사이의 호출관계에 의한 오버헤드가 생기지 않는다. NSI 미들웨어는 그 자체가 컴포넌트의 집합으로서 실장 된다. (그림5참조). 또, NexWave는, 얼마쯤 OS위에 NSI 컴포넌트들을 실장하고 있다.

### 5.2 디지털AV기기로 향한 NSI 기능 확장

하다찌 그룹은, “T-Integrator” ( $\mu$ Tron 위에의 NSI 미들웨어)를 이용해서 자사의 디지털텔레비전의 소프트웨어를 컴포넌트화 하였고, NSI의 기능, 성능 등을 평가했다. 그 평가 결과들을 근거로 해서, 디지털AV기기의 기반소프트웨어에 필요한 아래의 기능을 개발하고, NSI 기능들을 확장하였다.

- (1) 컴포넌트들을 복수의 메모리디바이스와 메모리영역에 배치할 수 있도록 하는 멀티 셕션기능
  - (2) 파일시스템을 갖고 있지 않는 제품에, 교체용 컴포넌트의 배치가 가능하도록 하는 골드 스와핑의 기능 확장
  - (3) ICE(In-Circuit Emulator)을 이용한 컴포넌트의 디버그 기능
- 이러한 기능 확장에 의하여, 디지털AV기기에의 NSI의 적용이 처음으로 가능하게 되었다.

### 5.3 컴포넌트 관리 기구

컴포넌트 관리기구의 목적은, NewTron 위에서 개발하고, 축적되어 왔던 과거의 막대한 양의 소프트웨어 자산을 Linux 위에서 그대로 활용할 수 있도록 하는 일 과, 장래에 하드웨어의 진화에 대비하여 유연하게 대응할 수 있도록 소프트웨어 컴포넌트의 골격을 제공하는 일이다.

컴포넌트 관리기구는, Linux의 미들웨어로서 동작하고, Linux로 동작하는 소프트웨어를 컴포넌트화 한 것이다(그림6참조). 더욱이, 그림6의 "Lib TRON"의 기능에 의해, T-Integrator위에서 동작하는 컴포넌트들을 Lib TRON위에서 동작 시킬 수 있는 일이 가능하게 된다. Lib TRON에서는, 시스템 콜 변환이야 NewTron 위에서의 타스크의 상태관리 등, NewTron위에서의 컴포넌트가 정확하게 동작하기 위해 필요한 리소스 관리기구를 제공한다. 이것에 의해, NewTron 위에서 개발, 축적해 왔던 소프트웨어 자산을 적은 비용으로 Linux에 이식시키고, 새로운 Platform에서 활용할 수 있다. 또, 그림6의 프록시(처리대행프로그램)으로, 컴포넌트화한 소프트웨어하고 Linux의 네가티브한 환경에서 동작하는 어플리케이션과 라이브러리 등(비 컴포넌트화한 소프트웨어)의 커뮤니케이션 기능을 제공하기 때문에, 기존 소프트웨어 자산을 Linux위에서 새롭게 개발한 어플리케이션에 통합하는 일이 가능하게 된다. 기존 소프트웨어 자산의 이식성에 관해서, 디지털텔레비전의 소프트웨어를 예로 시험한 결과, 대상 소프트웨어규모의 1~3% 의 개발규모로 다른 Platform에의 이식이 가능하게 될 것이라는 전망을 하고 있다.

컴포넌트화한 소프트웨어는 하드웨어 부품처럼 일체화시켜서, 블랙박스기술로서 개발시키는 히타찌그룹 제품의 강한 원천 기술이다. 한편, OS에 Linux를 채용하는 것에 의하여, BUILDER시스템(build & compile system)에서도 외부의 리소스를 활용하여 개발의 스피드업을 꾀하는 것이 가능하게 된다.

이와 같이, 수직 종합형과 모듈라형의 장점을 모두 경비한 개발기술과, NSI의 우수한 특징을 살린 소프트웨어의 재이용 기술에 의해, 디지털AV기기의 개발 효율은 비약적으로 향상된다.

---

## 6. 끝으로

현재 상태의 Platform에서는, MPU하고 IP을 자유롭게 편성해서 시스템LSI를 구성하기 까지에는 아직 이르지 않았고, 소프트웨어 컴포넌트의 부품화의 완성도도 아직 이르다. 그러나, 이것의 NSI 기술의 Platform을 통해서, 지금까지 개발해 왔던 광대한 디지털 AV 기기의 소프트웨어 자산과, Linux의 위에서 개발한 홈 네트워크의 미들웨어와의 통합에 의해, 디지털 가전과 IT분야의 기술 융합의 효과가 확인되고 있다. 앞으로 점점 중요하게 되어 가고 있는 네트워크에의 응용과 지적 콘텐츠의 관리 등등의 개발에서는, 사내 사외를 막론하고 오피쇼우 개발을 포함한 최적한 개발체제를 구축하는 것이 가능하게 되었다고 생각 한다.

시스템 LSI와 편집 소프트웨어의 새로운 개발 수법과 기술의 연구에서는, 종래, 그 연구 성

과를 적용하기 위해 개발의 현장에서 커다란 노력을 치를 필요가 있었다. 선단, 기반연구의 성과를 Platform에 활용하는 최초의 예는, 모델 구동 개발 수법 등의 소프트웨어 공학에 근거한 새로운 개발 수법, 그리고 기술의 개발 현장에의 도입이다. 모델 구동 개발 수법의 확립에는 시간을 필요로 한다. 그러나, 타사에 앞 질러서 실용화하고, 업계 텁의 개발 스피드를 실현할 생각이다.

히타찌 그룹은, 그룹 각 사가 협력해, 최종 목표 달성을 지향해서, 위에 언급된 능력을 가진 Platform 개발을 계속해 나가고 있다.

## APPENDIX

표1 NSI의 기능 개요

데이터와 코드의 프로그램화, 다이나믹 로딩, 분산 처리 기구 등, 종래의 편성소프트웨어에서는 실현이 곤란했던 기능을 제공한다.

기능	개요	
캡슐화 (컴포넌트화)	모듈의 이용은 명시적으로 정의 된 인터페이스에서 밖에 허가되지 않는다.	
컴포넌트 구성 관리	컴포넌트 사이의 미리 정의되어진 인터페이스에 근거해 서 연결해, 관리하는 기능	
컴포넌트 실행관리	로드타임관리	시스템 기동 시, 필요한 컴포넌트를 로드하고, 참조 연관 관계를 완결하는 기능
	런타임 관리	컴포넌트 실행 시, 동적으로 로드 하는 기능
다이렉트 바인딩	컴포넌트 사이의 호출을, 관리 컴포넌트를 개입시키지 않고 직접 실행하는 기능	
분산처리기구	복수의 MPU에 대한 컴포넌트 실행 관리 기능	

그림 5 NSI의 실장에 넥스웨이브 솔루션 사는, NEXCORE랑  $\mu$ Tron을 시작으로, 얼마쯤의 OS위에 NSI를 실장하고 있다.

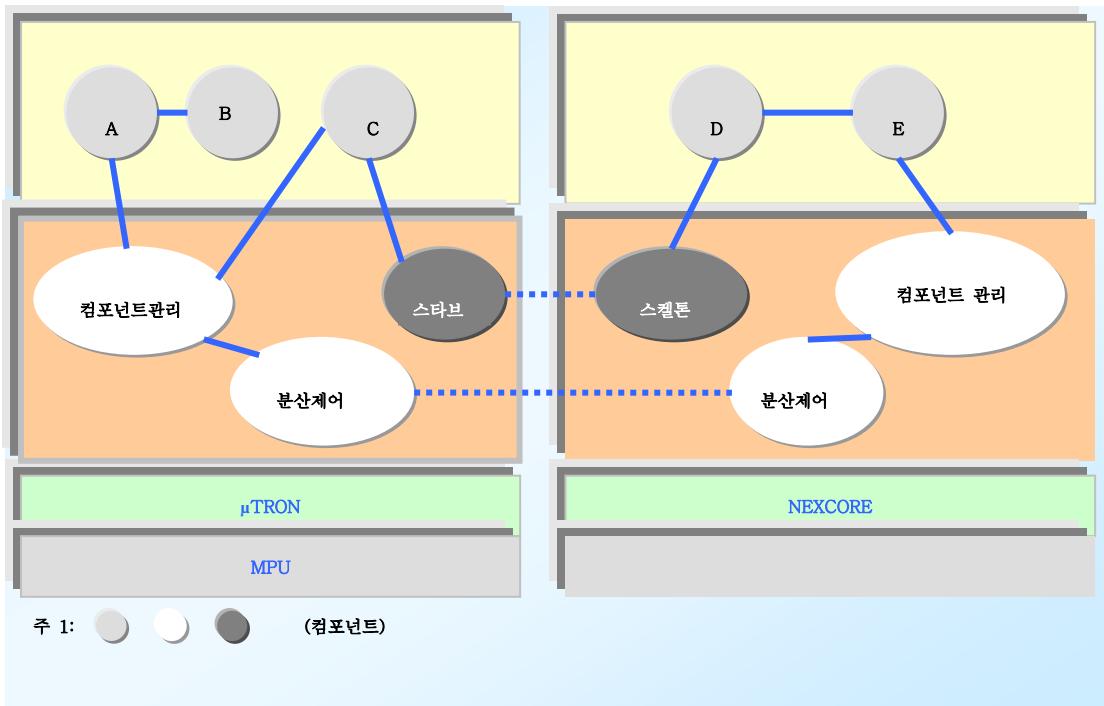


그림 6 컴포넌트 관리기구: 뉴트론 위에서 개발한 기존의 소프트웨어 자산을 리눅스에 의식하고, 하드웨어의 변화에 유연하게 대응할 수 있는 컴포넌트화의 골격을 제공한다.

