PHILIPS

sense and simplicity

Accelerating Digital Television Innovation

Introducing jointSPACE

Bas Engel
Philips Consumer Lifestyle - Business Unit TV
ELC Grenoble, October 15, 2009

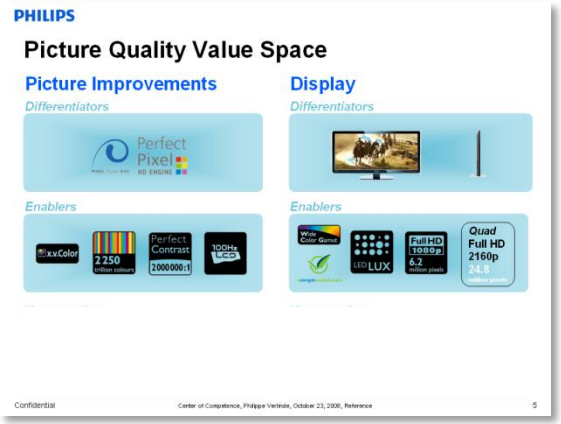
Outline

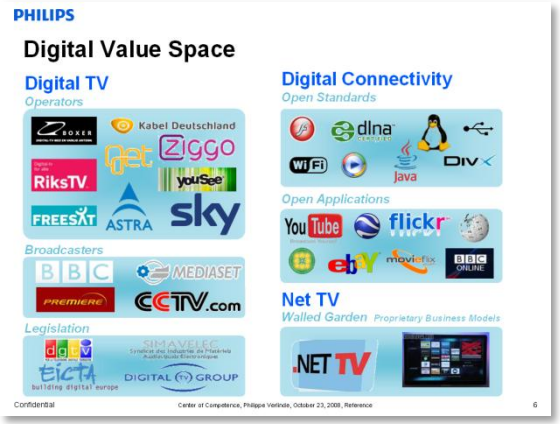
- Changing digital TV landscape
  - Managing software complexity
  - Leverage standard software assets
  - Required industry framework
- Accelerating innovation
  - Independent software asset development
  - Extending system capabilities
  - Enhancing embedded development

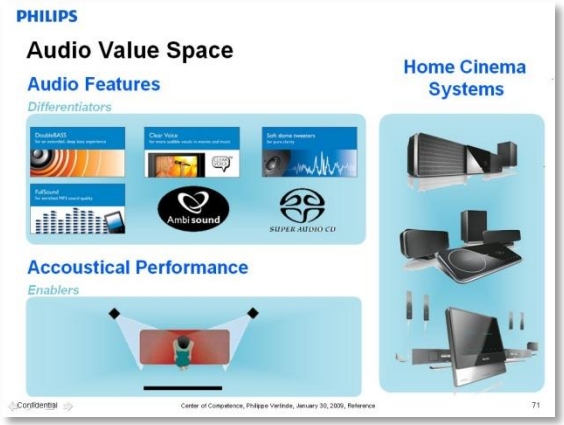
PHILIPS

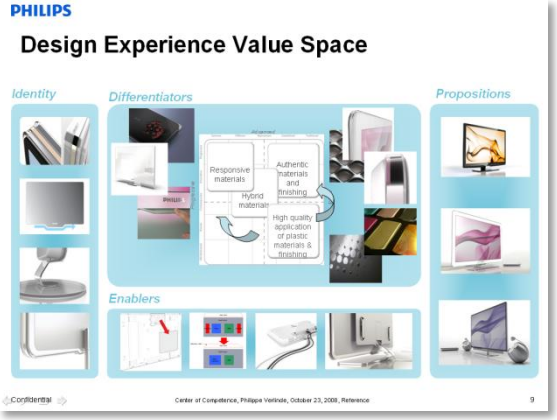
joint space
split application architecture

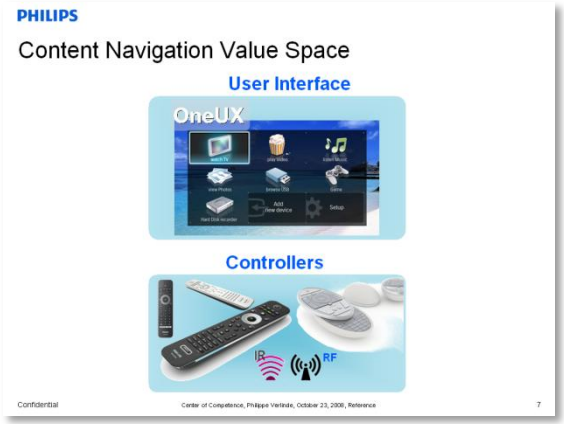
TV Value Spaces







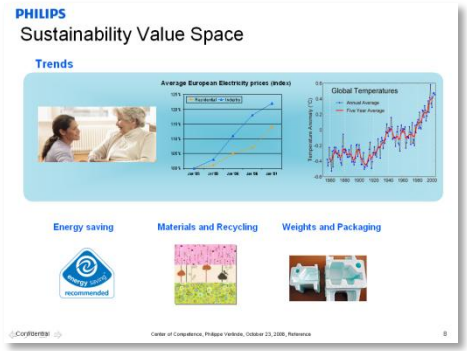


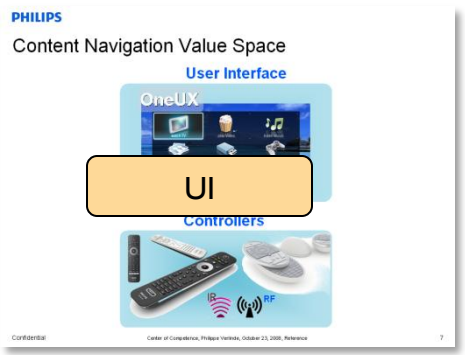


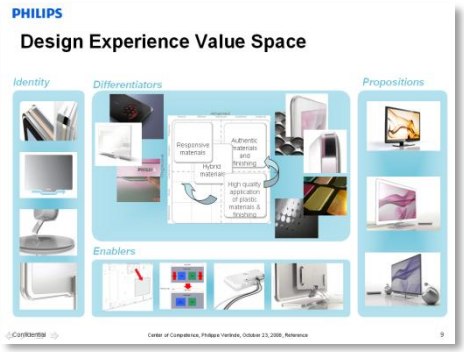
PHILIPS

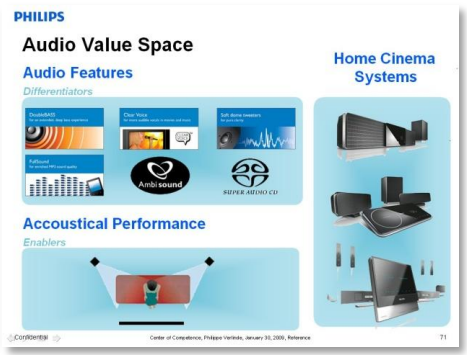
joint space
split application architecture

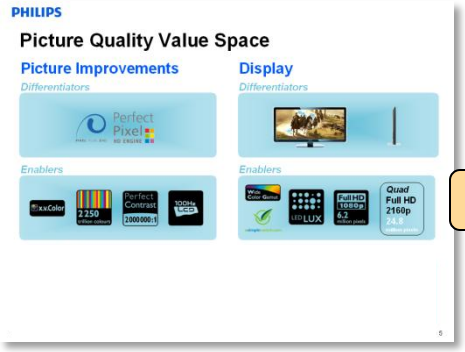
Software Driven Value Spaces

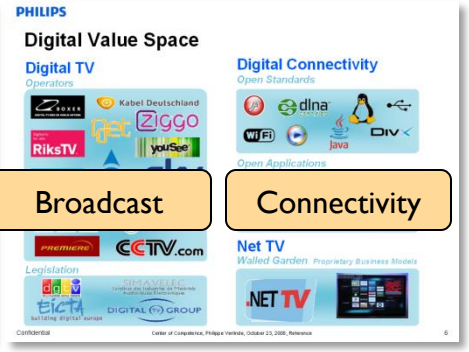










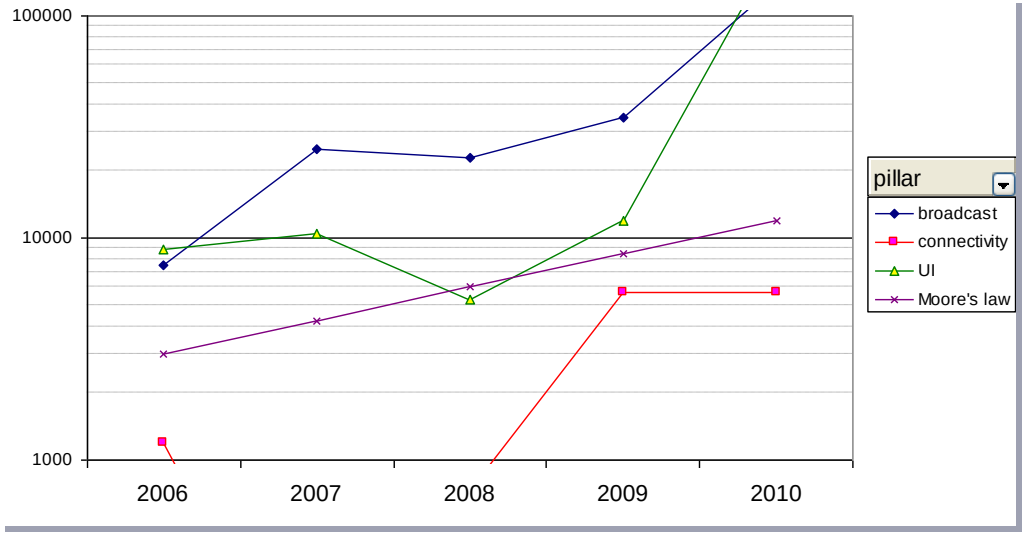


Supporting — Leading

PHILIPS

joint space
split application architecture

Software Value Space Growth



Code

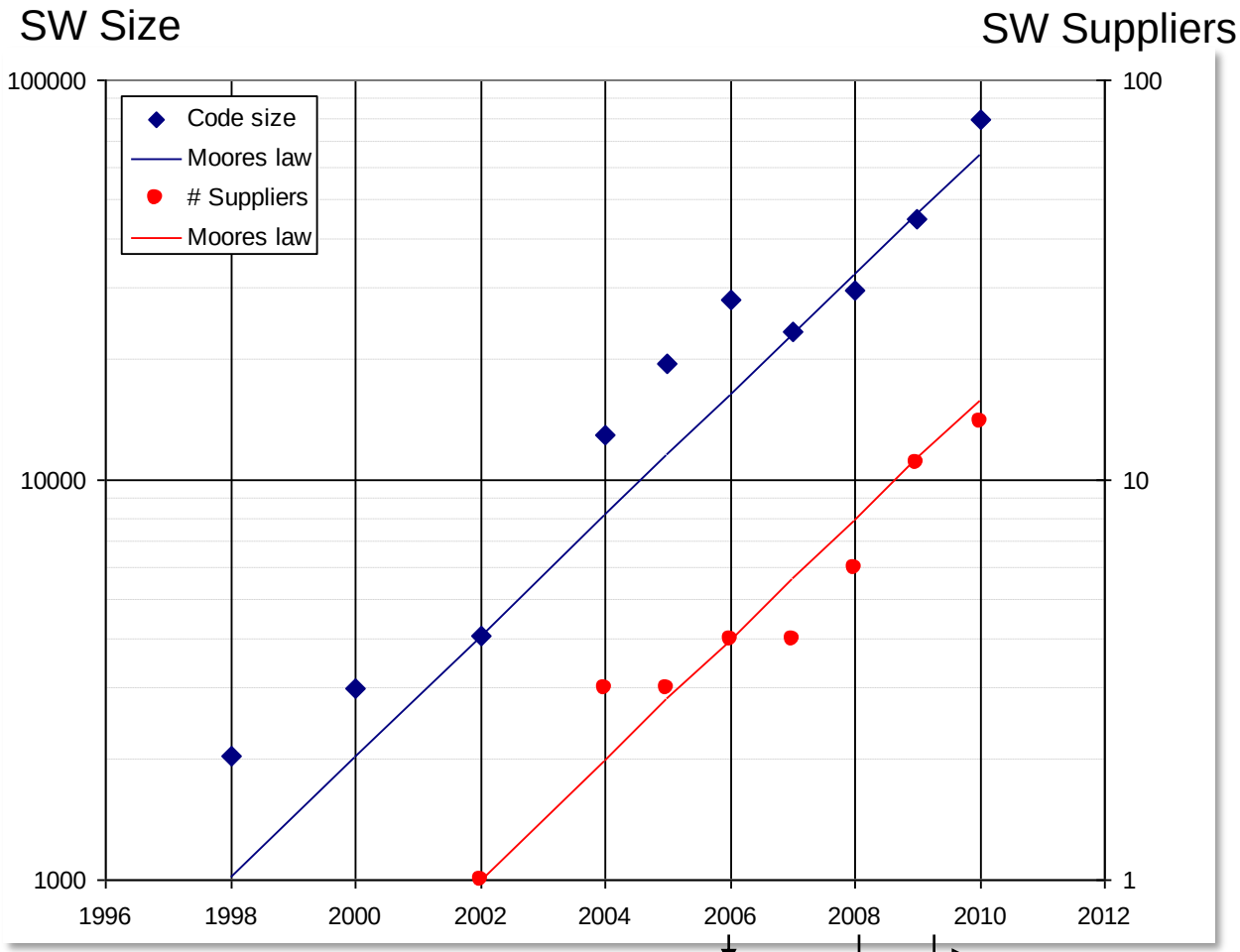
RAM

Software value spaces accelerates beyond Moore's lay

PHILIPS

joint space
split application architecture

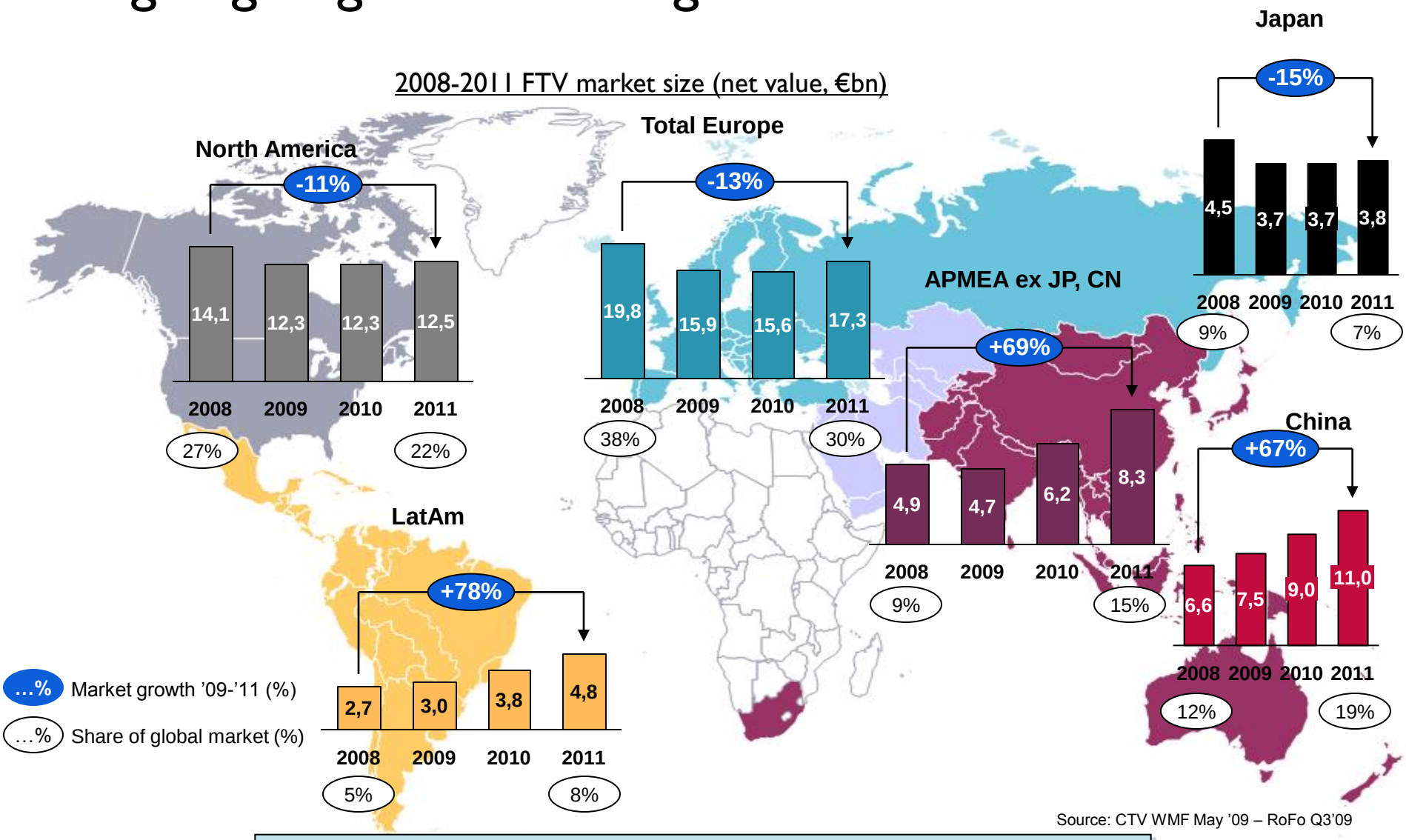
Software Featuring Dominates Growth



PHILIPS

joint space
split application architecture

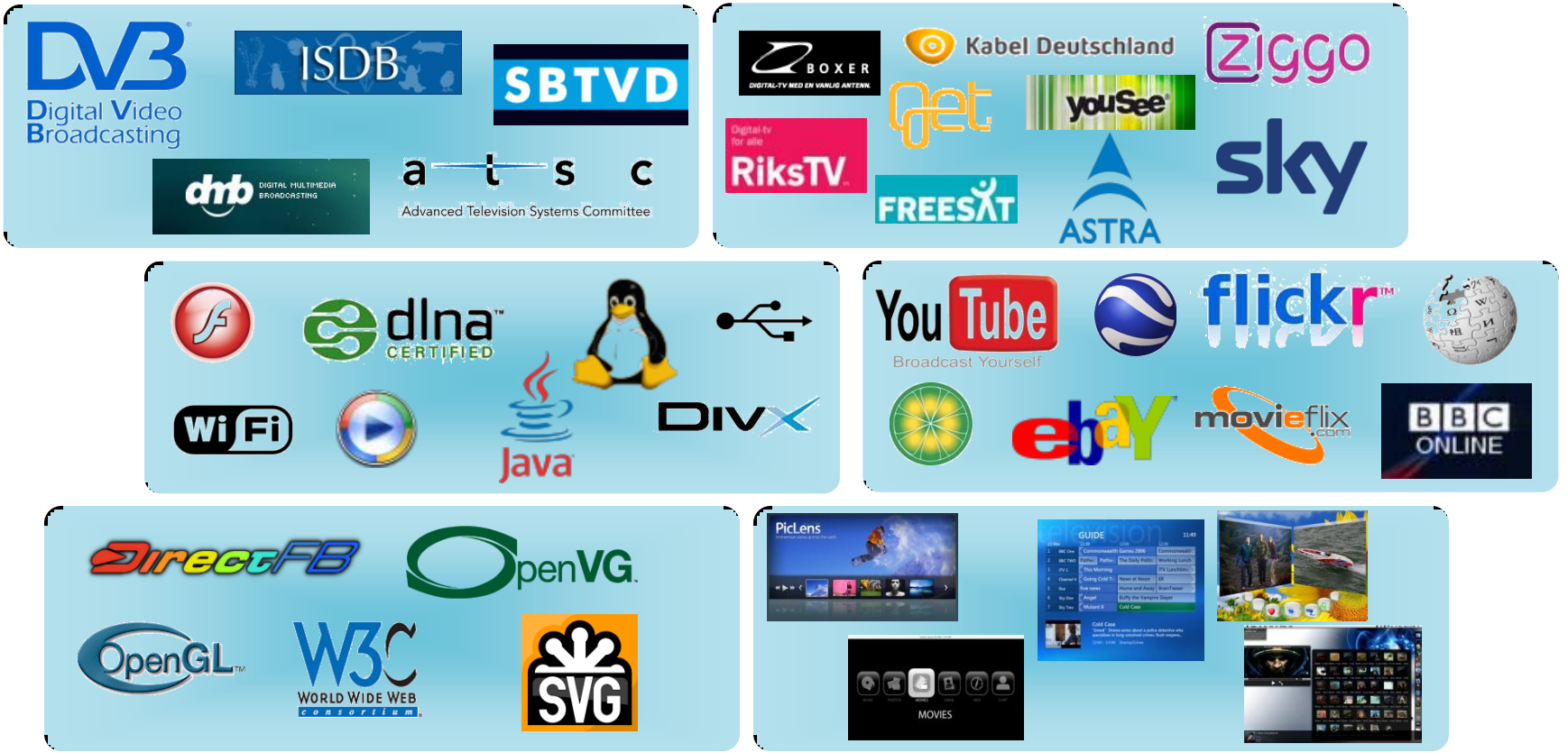
Ongoing Regional Challenges



Source: CTV WMF May '09 – RoFo Q3'09

Digital broadcast value space continues to expand

Ongoing Digital Featuring



Maximize Reuse of Development and Maturing Effort

PHILIPS

joint space
split application architecture

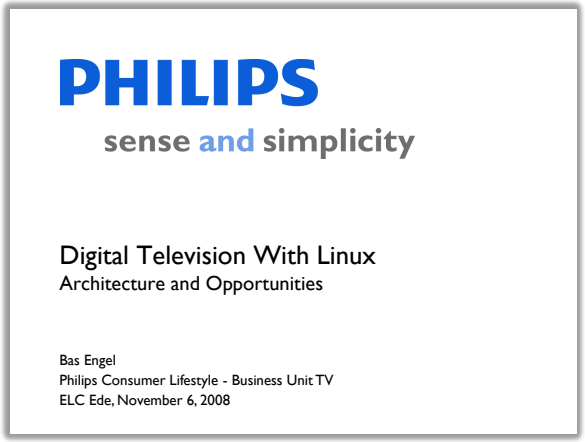
Managing SW Complexity And Diversity

- Had to change from highly integrated architectures to distributed ones
  - Fast and predictable integration of system extensions
  - Avoid an extensive (re)validation cycle

- Need to manage building blocks fully independently
  - Allowing independent software asset development
  - Limited asset correlation, cater for extensions without knowing all details

- Defined solutions for multi-client connection and resource management
  - AM sets destination, clients control source
  - Applications can request resources dynamically
  - Independent application lifecycle and execution

SPACE concept presented at 2008 ELC



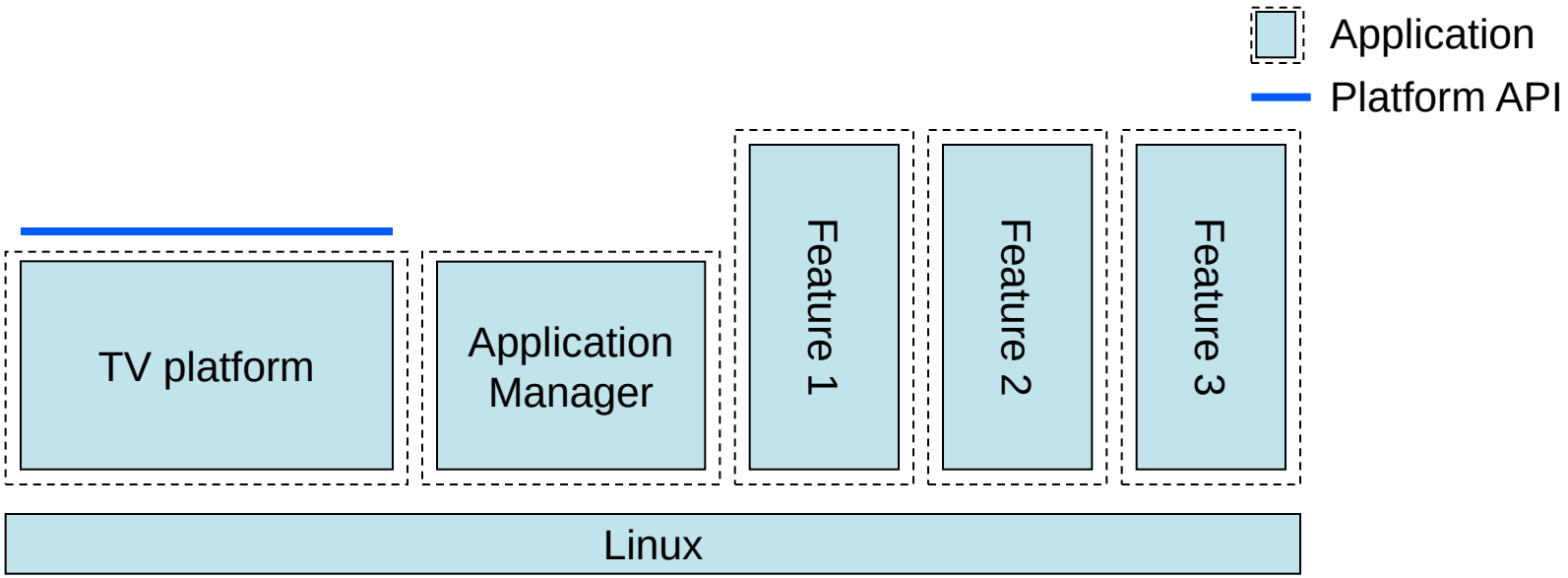
Industry Recognition

- All suppliers are facing ongoing demand to deliver software featuring
  - Internet browsers, DLNA, broadcast standards, UI improvements, etc
  - Closely following Moore's Law
- Both suppliers and A brands have a similar challenge in delivering this
  - Can no longer develop all IP in-house, must leverage standard solutions
  - Standard solutions that must fit a standard system context
- Standard system context requires some industry standard framework
  - No proprietary concepts and source code
  - Based on a industry framework, not a standard with compliancy rules

PHILIPS

joint space
split application architecture

Industry Framework Requirements



SPlit Application arChitecturE

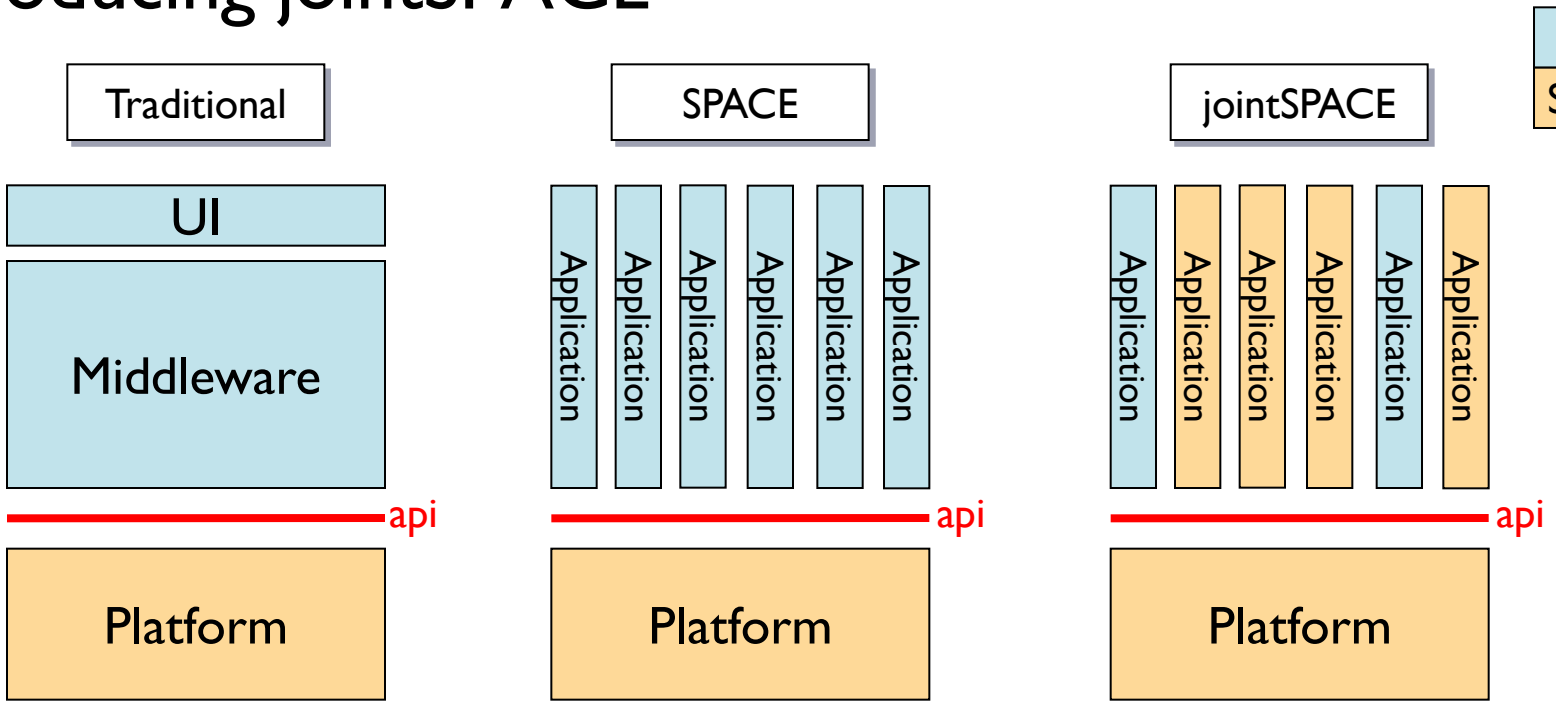
- The resources in the system are explicitly and centrally managed
- The client applications are system context unaware
- System integrity driven by centrally managed application lifecycle, focus, and visual layout
- Fast and predictable system extension (process architecture, libraries)

Application orthogonality
System behavior must be independent of the application composition

PHILIPS

joint space
split application architecture

Introducing jointSPACE



- 'Traditional' platform only engagement not feasible anymore
  - Both for suppliers as well as for PCL
- 'SPACE' enables required architecture flexibility
  - Still fully PCL system integration responsibility
- 'jointSPACE' drives extensibility and supplier leverage
  - Customer (Philips) differentiation on top of standard system

Win-Win: Supplier Value Space

- Joint engagement on system context increases supplier value
  - Philips Consumer Lifestyle (PCL) engaged to mature supplier assets
  - Standard supplier assets can be reused for non-PCL engagements
- Lower threshold for additional TV customers to engage with supplier
  - Proven system maturity with supplier reference solution
  - Optimal cost spreading across customers
- Drives supplier to create top performance end-2-end systems
  - Enables pro-active plan by the supplier for system performance
  - Customer investments driven by customer specific innovation

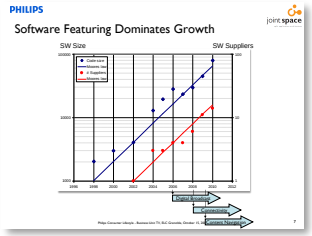
Supplier value space is a system partnership

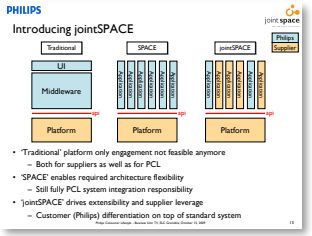
PHILIPS

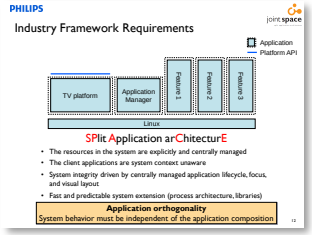
joint space
split application architecture

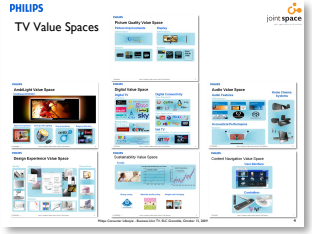
Changing Digital TV Landscape

- Continued software growth requires new engagement model
  - Develop systems as a joint responsibility
- Joint engagement increases overall value
  - Mature supplier assets, leverage SW investments
- SPACE gives necessary control points for customization
  - Recognized in the industry by leading TV platform suppliers
- Enables TV manufacturer to focus on differentiation
  - Accelerating innovation





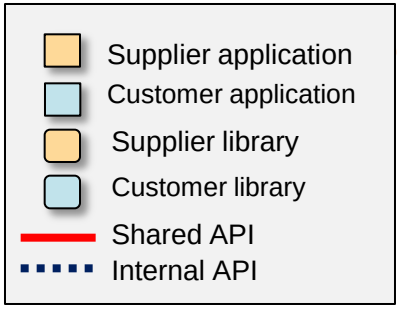




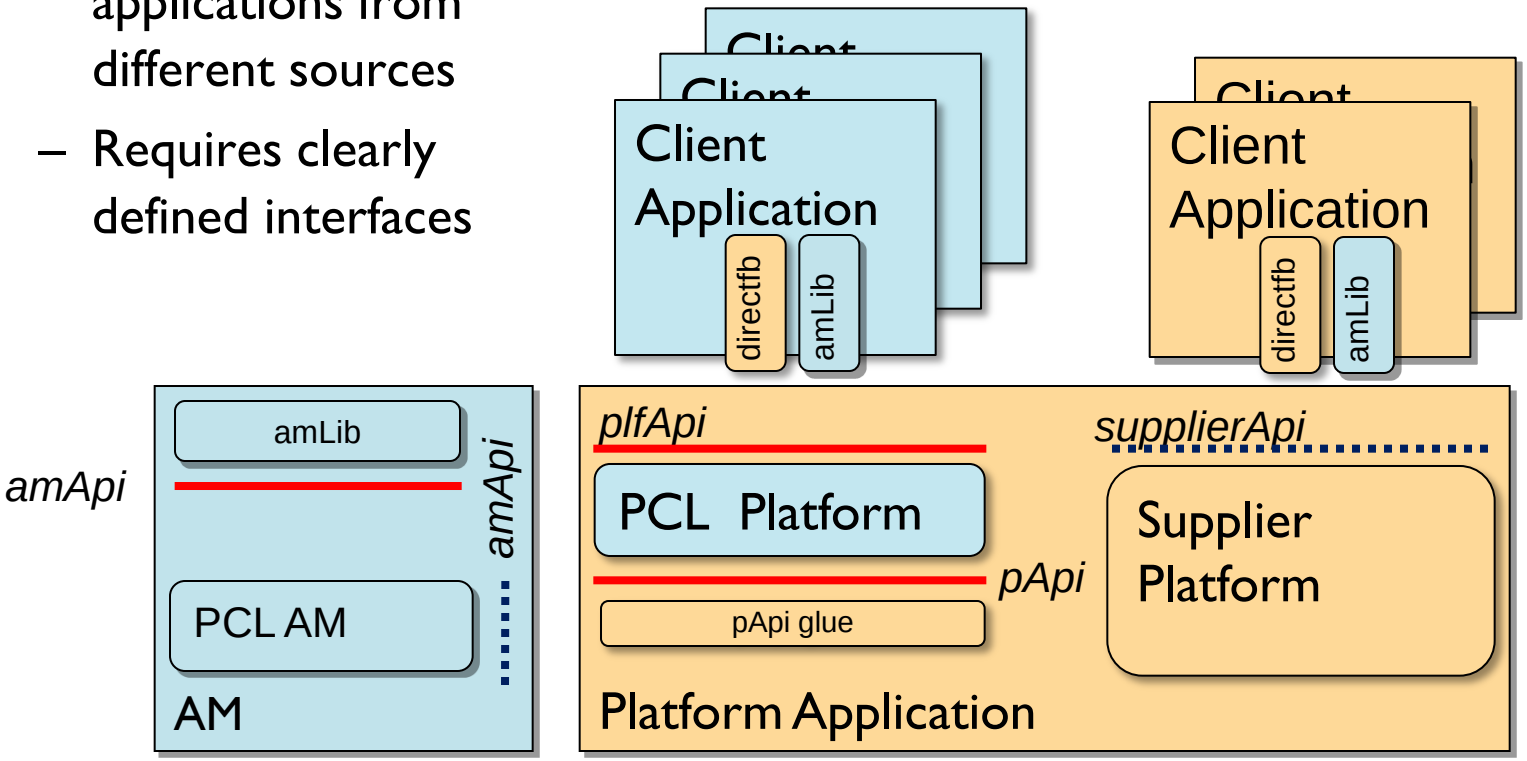

Accelerating Innovation
Independent software asset development, Extending system capabilities, Enhancing embedded development
PHILIPS

PHILIPS

jointSPACE Engagement

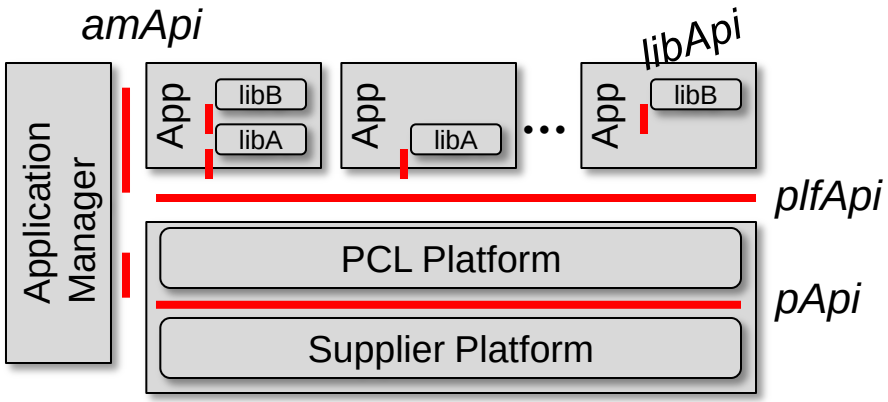


- Infrastructure provided by supplier (Linux, DirectFB)
- Supports applications from supplier and customer
  - Need to support applications from different sources
  - Requires clearly defined interfaces

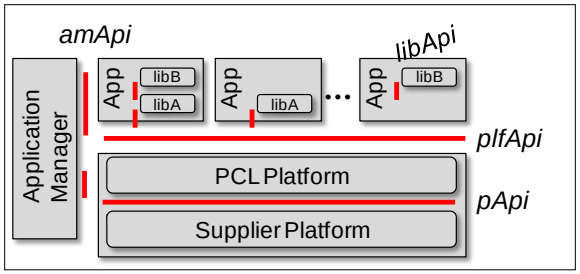


jointSPACE Porting API

- APIs between SPACE building blocks are called the "Porting API"
  - amApi: by client applications to make requests to application manager
  - plfApi: by client applications controlling the platform application
  - pApi: internal API between PCL plfApi and supplier platform interface
  - libApi: for data sharing across client applications

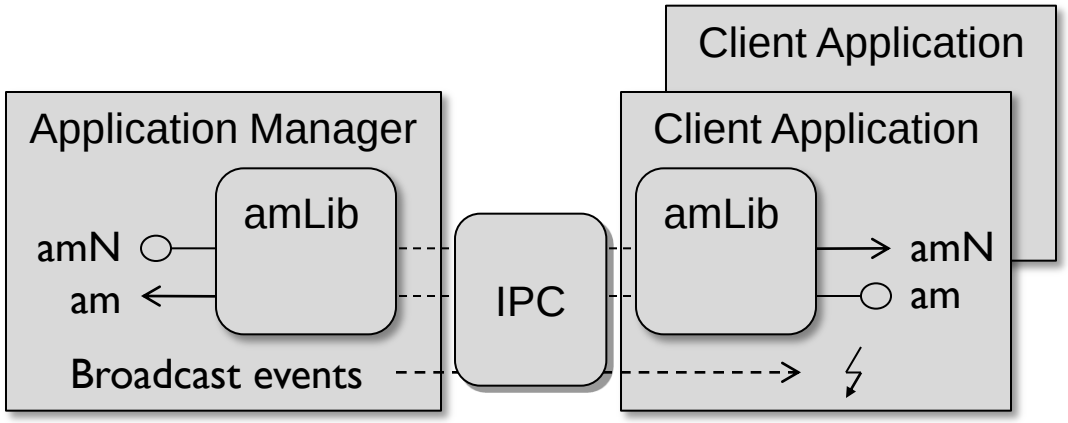


amAPI



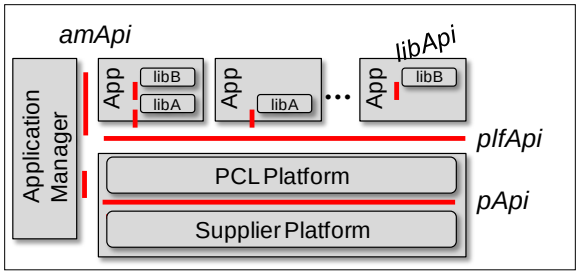
- Defines the interaction between Application manager and applications
  - Consist of a functional API and broadcast events
  - Asynchronous in both directions

- The Application Manager can address individual client applications
  - The Linux process ID (PID) will be used for this
  - The library (amLib) enables event broadcasts via IPC



PHILIPS

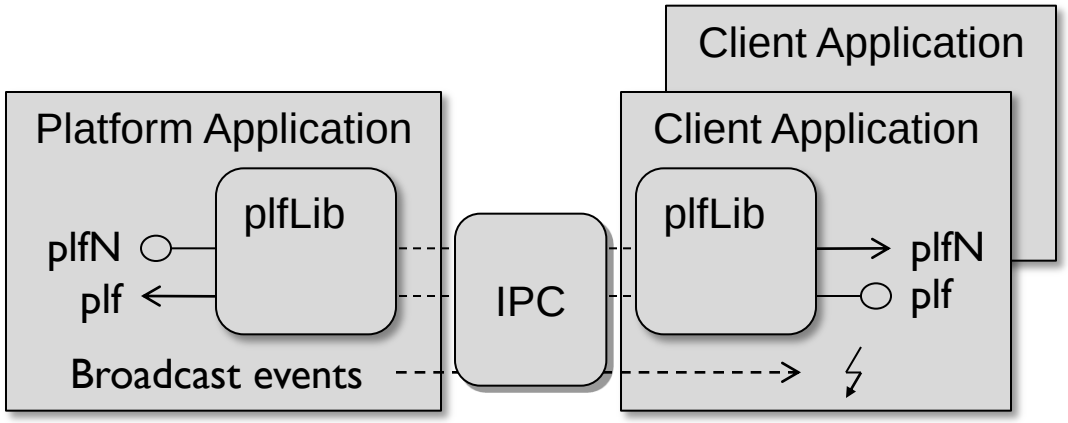
plfAPI



joint space
split application architecture

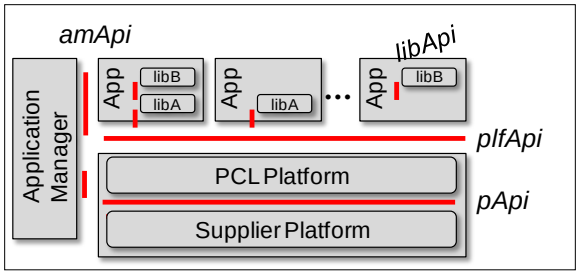
- Defines the interaction between Platform Application and applications
  - Consist of a functional API and broadcast events
  - Can be synchronous from application to Platform Application
  - Always asynchronous from Platform Application to applications

- The library (plfLib) enables broadcast events via IPC
  - During initialization IPC connection is established



PHILIPS

pAPI



joint space
split application architecture

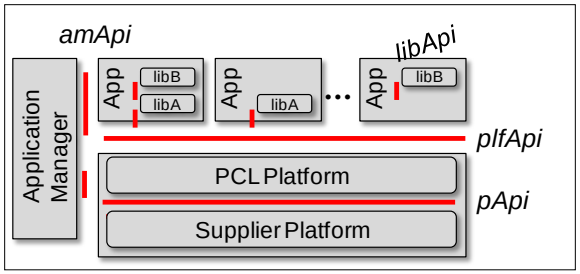
- pApi defines the AV API as the platform porting layer
  - Derivative of plfApi (without resource management ID)
  - Must be implemented by suppliers

- Internal to Platform Application
  - No client application can access this API
  - Suppliers are the only users

- Must be used within process boundary
  - To avoid overload of IPC traffic between pApi and plfApi
  - Hence supplier glue and PCL platform code always in single process

PHILIPS

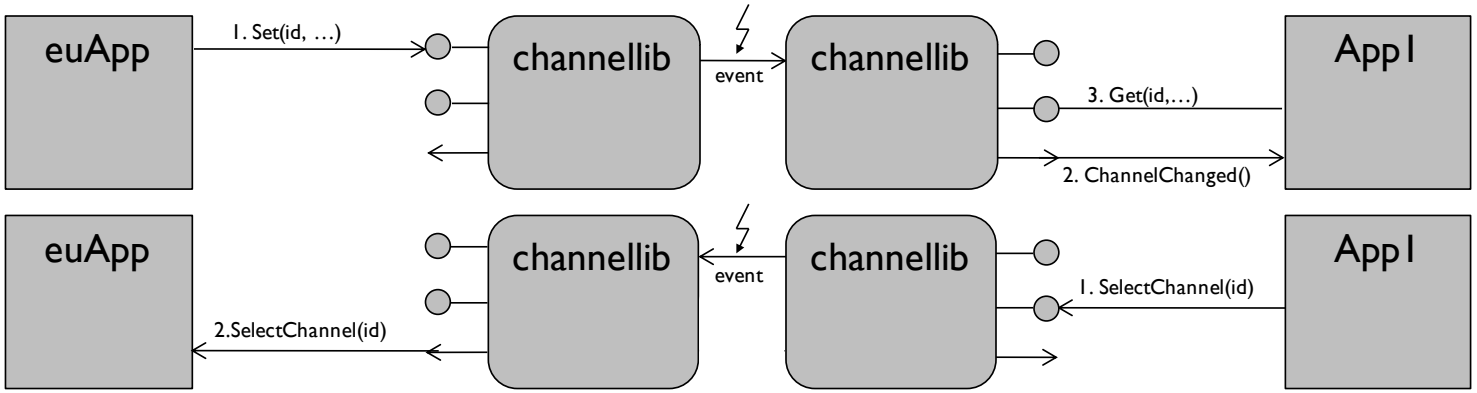
libApi



joint space
split application architecture

- General concept of SPACE is that applications do not communicate
  - Any application can gather data needed by other applications
  - Data must be shared, without creating application dependencies

- Concept used are (shared) libraries
  - Library relies on broadcast events to inform changes to applications
  - Provides a functional API, hiding communication channel



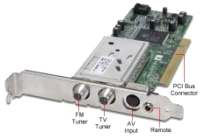
Porting Guide

- Detail description of SPACE framework
  - SPACE Concept, API Concepts, Application Lifecycle, Application States, Resource Management, Connection Management, Application Switch, Audio Video Control, Audio Video Platform Interfaces, Application Data Sharing

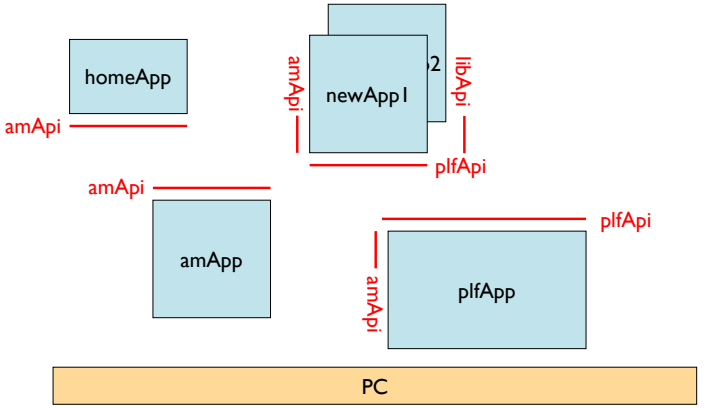
- All APIs documented and publically available
  - API behavior explicitly described, including detailed sequence charts
  - amApi, plfApi, pApi, libApi

Innovation Carrier

- PC enables fast prototyping of future TV use-cases
  - Enhanced multi-window
    - Compose multiple input streams
    - Real window rendering on PC
  - Enhanced graphics acceleration
    - OpenGL, SVG, Flash, DirectFB 2.0
  - All via PC cards and prototype extensions



- jointSPACE released on SourceForge
  - Sample applications
  - Complete documentation
  - SPACE fully available on PC using simulated platform



SourceForge Archive

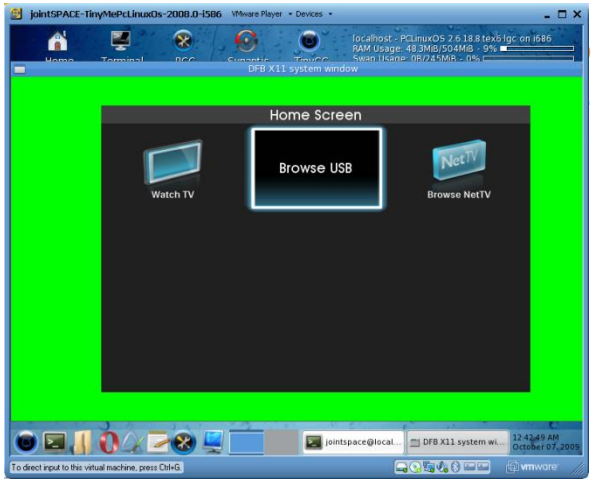
- jointSPACE on Linux/PC implementation available via SourceForge
  - Link http://jointspace.sourceforge.net

- The following can be downloaded
  - Porting Guide (pdf file)
  - HTML documentation for the APIs (amAPI, plfApi, papi)
  - Source Code package

- Current source code release focused on the APIs
  - Interface interaction and system dynamics

PHILIPS

Developing For jointSPACE



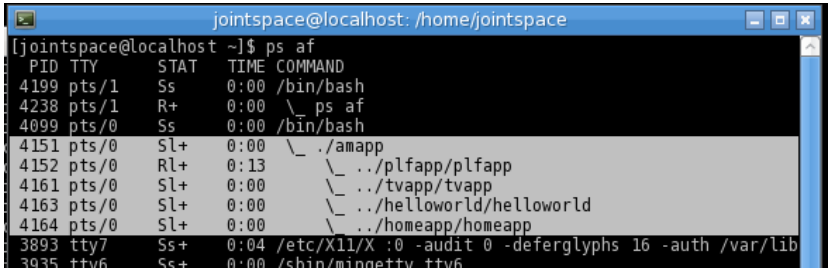
- Any recent Linux distribution expected to be usable
  - Validated for Ubuntu distribution
    - For “non-standard” Linux system, check steps in install.sh
  - Specific version of DirectFB, FusionDale and SaWMan required
    - See directfb/directfbinstall.txt for instructions

- Available are executables and libraries with required header files
  - Application Manager; Hello World, Home, Platform, and TV Application

- Each executable has a makefile
  - Using the file common.mk from the root
    - Two targets are supported: <appname> and clean
  - The BASE environment variable needs to be set correctly
    - Adapt common.mk or run setenv.sh

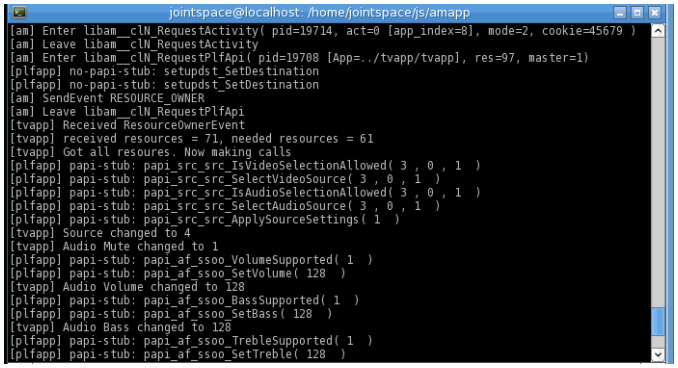
Simple Starting Point

- HomeScreen application is available
  - 'm' toggles HomeScreen, Navigation via cursor key, selection via Enter
  - In Watch TV (tvapp), the 'o' toggles a menu with calls to plfApi



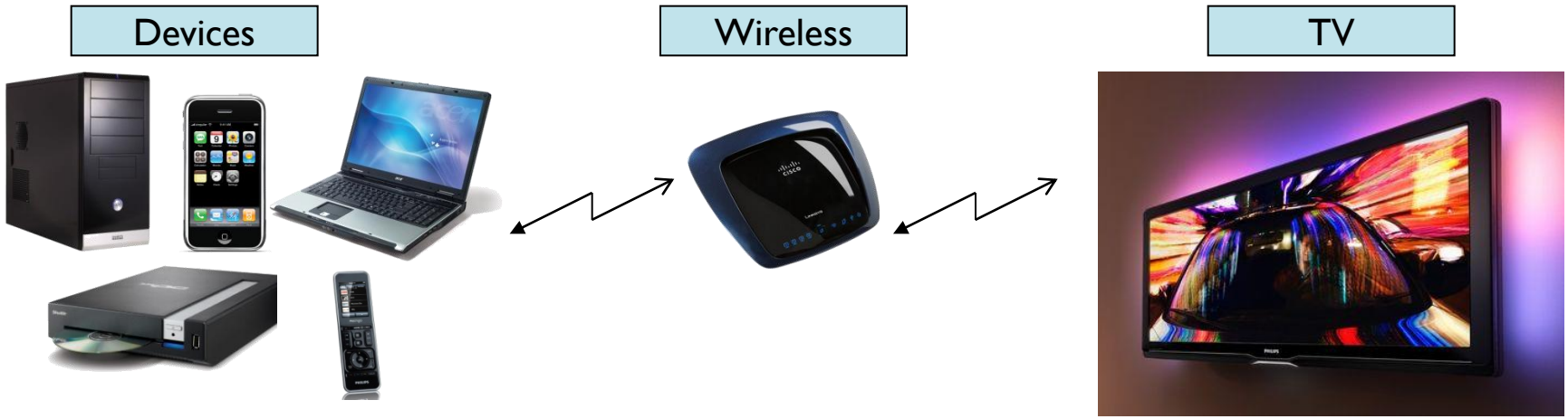
- amApp that manages the applications
  - Configured via application.lst file
  - Subset of amApi is implemented
  - Application launch and switch, focus switch, resource management

- While running console output is generated
  - Giving insight in which calls are executed.
  - Indicates program that generated the output



Accelerate Embedded Development

- Will enable remote access to TV via IP network connection
  - plfApi, amApi, libApi will become (partly) remote accessible
  - APIs are remotely accessible via xml-rpc
  - Leverage TV platform for AV decoding and rendering



- Run TV related applications on other devices
  - Using available interfaces to develop new applications
  - E.g. an Android application interacting with the TV

Shorter Lead-Time of Innovations

- Applications on PC in general needs to be constrained running on TV target
  - Unrestricted access to system resources cannot be granted
  - Ensuring system stability, performance and DRM restrictions

- Application must be constrained using a number of techniques
  - Disallow super-user power (remove POSIX capabilities)
  - Fine-tune system access using SMACK
  - Constrain memory usage via cgroups
  - Restrict CPU usage (e.g. via cpulimit, cgroups, or priorities)
  - SMACK for Digital TV discusses some of the technologies required
    - Paper by Embedded Alley
    - Presentation at this conference

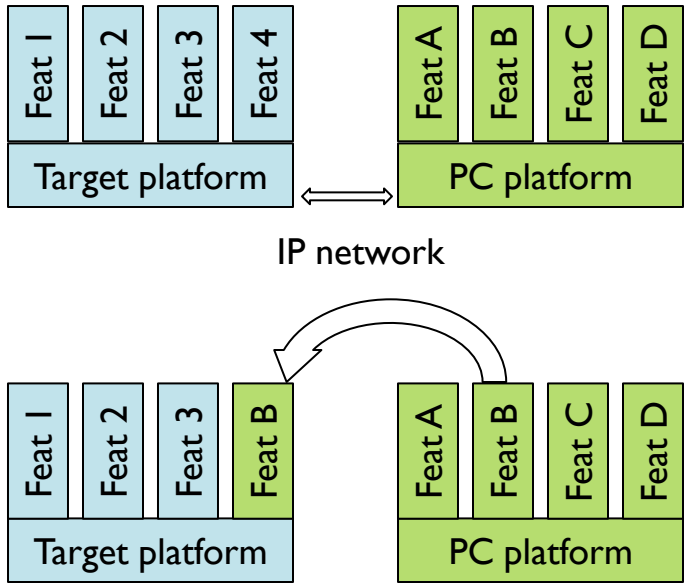
DirectFB 2.0 Further Enhances SPACE

- Enhanced application and window management
  - Animated application (window) transitions
  - Extended sub window support (multiple depth levels)
  - Support for different graphics APIs (e.g. OpenGL, Native 2D, …)

- Enhanced distributed UI capabilities (Voodoo)
  - Drawing down by an external device, final rendering done by TV
  - Window creation must be aligned with TV based amApp
  - Enables the use of different graphics cores for various tasks

Distributed TV Development

- Features developed in jointSPACE PC framework can be made available on TV
  - Framework on target and PC are identical (same API)
- Applications will be able to control target remotely
  - Using IP network
  - Requires specific pairing with TV
- Applications can be deployed on target directly
  - Requires recompilation only
  - Constrained by target resources

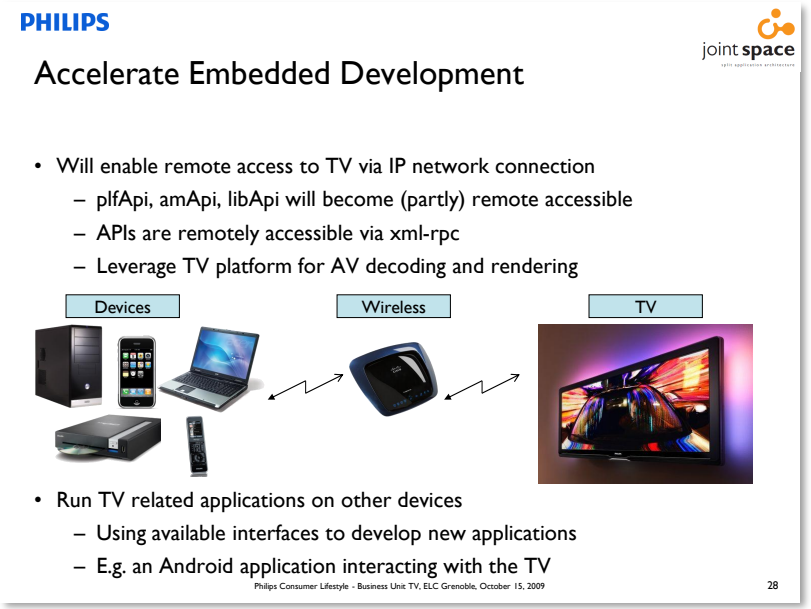


Enabling jointSPACE

- Philips internet TVs will be jointSPACE enabled in due time
  - Allowing applications to be developed outside TV target (e.g. using PC)

- Enabling others to develop new applications
  - And experience them on the actual TV

- SourceForge archive will be extended
  - With remote TV access
  - Available application interfaces

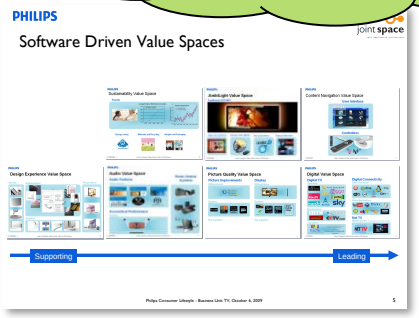


PHILIPS

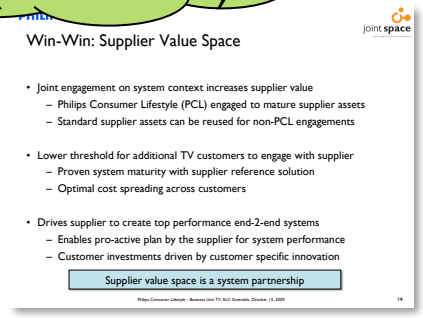
joint space
split application architecture

Driving Open Innovation

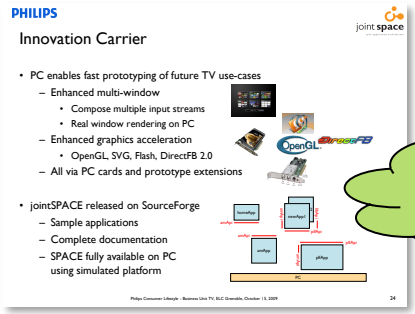
Philips



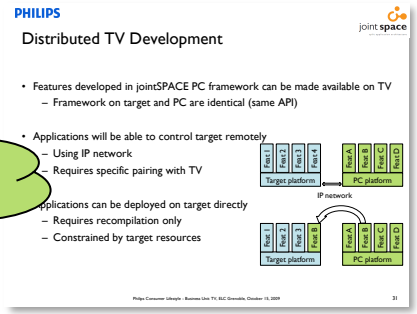
Suppliers (HW & SW)



jointSPACE

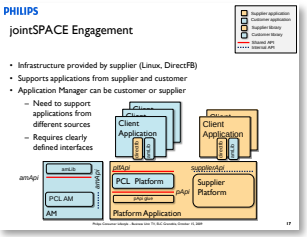


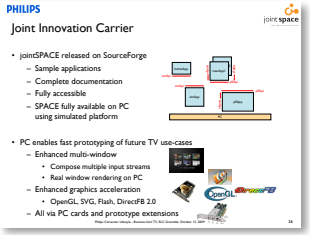
Innovators

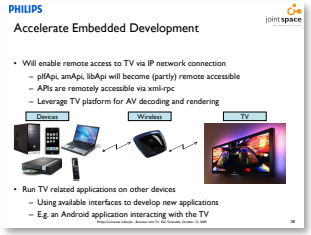


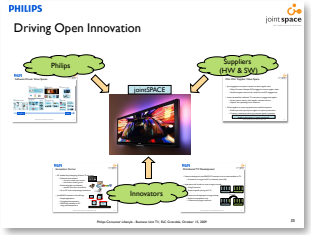
Accelerating Innovation

- jointSPACE engagement requires clear framework description
  - Enabling independent asset development
- SPACE framework used to create a new opportunity
  - Public PC environment to develop new applications
- Digital TV innovation must be made accessible and usable
  - Enabling the TV for external application development
- Towards open innovation
  - Working with an community of innovators










PHILIPS