

Example Scenarios: Scalable Media Server

- Simplest scenario – Scalable Media Server:
 - Stores conformant scalable media
 - Accepts client requests with Capabilities & Preferences
 - Transcodes content appropriately and delivers to client directly

Scalable Media Server does not need to know or care what the media-type is.

Generic Scalable Media Server

SCISM 31

Example Scenarios: Edge Servers for Transcoding

- Clients connect to Edge Servers to receive content
 - Specifies their Capabilities & Preferences
 - Edge Server transcodes content appropriately

Internet Backbone

Scalable Media Server

Edge Servers

SCISM 32

Scalable Bit-streams: Multi-Tier Scalability

Each has relative or absolute TOCs (not shown)

1-Tier scalability

2-Tier scalability

3-Tier scalability

Atoms

Note: Each atom can be either cumulative for true scalability or a complete new version for multi-version scalability

SCISM 13

Scalable Bit-streams: Example

- Image Compression
 - Wavelet decomposition
 - Subband-by-subband coefficient scan
 - Bit-plane by bit-plane encoding

EZW, SPIHT

Tier 1 (SNR)

Tier 2 (SNR)

Tier 1 (Spatial Resolution)

Tier 2 (Spatial Resolution)

JPEG-2000 LRPC

JPEG-2000 RLCP

SCISM 14

Conclusion: Summary

- A systematic methodology for universal media-type agnostic transcoding of scalable media, has been presented.
 - Universal meta-format and meta-data for scalable media proposed.
 - Encoding standards are less important in this paradigm
 - Meta-format is a loose format for all scalable encodings
 - Protocol for Capabilities & Preferences Specifications proposed.
 - Both individual media components, as well as combinations for an overall experience media, have been considered.

SCISM 33

Conclusion: Technology Promises

- Scalable media can revolutionize the world:
 - Seamless content adaptation and ease of use
 - Automatic explosion in use of media
- 'Media' not limited to images/audio/video
 - New scalable media will evolve – 3D, 4D
 - Encoding standards too slow to keep up
- The proposals in this talk promise:
 - To remove the infrastructure expense bottleneck.
 - To make adoption of new scalable media easy by removing reliance on encoding standards.
 - To make real all the benefits of scalable media.

SCISM 34

Scalable Bit-streams: Examples

- Video Compression
 - MPEG standards support several types of scalability.
 - Streaming Needs
 - They are hardly being used.
- Need to cater to new scalable media
 - Need ways to deliver

1-Tier (Interactivity)

2-Tier (Temporal)

SCISM 15

Scalable Bit-streams: Transcoding

- Transcoding comprises:
 - Chopping off rows and columns of the multilevel bit-stream from outer ends
 - No need to decode or decrypt
 - Encoding & encryption causality must be enforced.

Tier 1

Tier 2

Tier 1

Tier 2

Transcoding

SCISM 16

Conclusion: Sign Off

- Contact:
 - Debargha Mukherjee
 - Email: debargha@hpl.hp.com
 - Tel. 650-236-8058

SCISM 35

Scalable Bit-streams: Encoding & Encryption

- To make such transcoding work:
 - Encoding Causality
 - Encryption Causality

Tier 1

Tier 2

While Encoding this block, do not use any information from non-causal blocks, even if they appear earlier in the bit-stream.

Tier 1

Tier 2

While Encrypting this block, starting state of the generator is derived from the ending states of adjacent causal blocks.

SCISM 17

SCISM: Parcels & Components

- Media is organized in "Parcels"
 - Parcel – a transmission instance that provides a composite media experience.
 - Parcel can have multiple "Components"
 - Component – Defines encoding boundaries
- Example:
 - Parcel – An image with audio commentary.
 - Components – Image data, Audio data

SCISM 18

SCISM: Top Level Parcel Format

Contains Meta-data

- High-level description of parcel
- Dependencies between parcel components
- Individual Component Headers describing the components

Contains Media Data

- Actual encoded multi-tier scalable data for each Component

Parcel Header

- Number of Components
- A MxM matrix D describing dependencies
- Multi-tier scalable bit-stream for component indexed with TOCs

Component Header List

- Component 0 Header
- Component 1 Header
- Component M-1 Header
- Meta-data describing scalability structure of component

Parcel Data

- Component 0 Data
- Component 1 Data
- Component M-1 Data

SCISM 19

SCISM: Component Dependency

Component Dependency

- A MxM matrix D defines dependencies between Parcel components.

$$D = \begin{bmatrix} d_{00} & d_{01} & d_{02} & \dots & d_{0,M-1} \\ d_{10} & d_{11} & d_{12} & \dots & d_{1,M-1} \\ d_{20} & d_{21} & d_{22} & \dots & d_{2,M-1} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ d_{M-1,0} & d_{M-1,1} & d_{M-1,2} & \dots & d_{M-1,M-1} \end{bmatrix}$$

Diagonal element d_{pp} indicates if the pth component must be included

Non-Diagonal element d_{pq} indicates if the qth component must be included or excluded if the pth component is included or excluded

SCISM 20

SCISM: Attribute Distributions

Distribution Specifications

- Exact
- Marginals – Distribution is product of marginals.

Example

- JPEG2000 (LRCP)

Attribute: Bandwidth (BW)

1.00	1.00	1.00	1.00
0.50	0.50	0.50	0.50
0.25	0.25	0.25	0.25

Attribute: Display Resolution (DR)

1.00	1.00	1.00	1.00
0.50	0.50	0.50	0.50
0.25	0.25	0.25	0.25

SCISM 25

SCISM: Parcel Data Format

Parcel Header

Parcel Data for Types D/I

Component 0 Data **Component 1 Data** ... **Component M-1 Data**

Component Data with Absolute Indexing

Component Data with Relative Indexing

Encoded and/or Encrypted Bit-stream Atoms

SCISM 26

SCISM: Component Header

Component Header

- Number of Tiers
- Number of Layers in each Tier
- Consistency
- S/M
- N
- Attribute Data List

Attribute Distribution

SCISM 21

SCISM: Attributes – A Crucial Bridge

Component attributes play a crucial role:

- They bridge the communication gap between:
 - The media Creator/Originator (Media meta-data) and,
 - The media Consumer (Capabilities & Preferences Specs).
- Attributes are:
 - To the **Media Creator**, quantifiable properties of media based on which one can transcode.
 - To the **Media Consumer**, quantifiable properties to indicate its limitations and preferences.
 - To the **Transcoder**, just numbers, based on which it must determine how to transcode a scalable bit-stream.
- Attributes are quantified by non-negative values called Attribute Values.

Attribute Examples

- Bandwidth
- Display Resolution
- Processing Power
- Number of Speakers
- Number of joystick buttons

SCISM 22

SCISM: Capabilities & Preferences: Use of Attributes

- Clients indicate their capabilities and preferences using attributes.
 - Reserved attributes are standardized
 - Custom attribute codes known to sw/hw system used for experiencing the media
- Transcoder needs to interpret the capabilities unambiguously.

SCISM 27

SCISM: Capabilities & Preferences: Constraint Definition Language

- Clients specify constraints on definable functions of attributes, called *Measures*.
 - Measure Definition:

$$M(\bar{a}) = \sum_{i=1}^N \lambda_i \prod_{j=1}^{n_i} f_{ij}(a_{ij})$$
 - N = Number of product terms
 - n_i = Number of elements in ith product
 - λ_i = Multiplier for ith product
 - $f_{ij}(x)$ = Simple univariate functions like $x, 1/x, \log(x), e^x$ etc. with codes standardized in the protocol
 - a_{ij} = Attribute codes
- Constraints:
 - Limit Constraints – Upper limit, Lower limit, Range of measure values
 - Optimization Constraints – Maximize, Minimize measures

Constraint Examples

- BW < 300
- DR > 1280
- Min(MSE + λ .BW)

SCISM 28

SCISM: Attribute Codes

- Each attribute is identified uniquely by a code (32-64 bits long)
- Types of Attributes
 - Reserved:** Universal across media-types (Bandwidth, Display_Resolution etc.)
 - Custom:** Companies buy *Attribute Code Space* to denote properties of custom media they create
- Designated field in Attribute Code indicates how to combine attribute values for 2 or more media components.
 - Additive, Maximum, Minimum, Multiplicative etc.

Attribute Combinations

For media with several components, transcoder can jointly optimize with respect to attributes common to two or more components such as bandwidth.

SCISM 23

SCISM: Component Attribute Data

Attribute Data

- Attribute Code
- Monotone Type
- A_{ref} (Reference Attribute Value)
- Attribute Distribution over atoms

Multiplier of A_{ref} to yield Attribute value for all transcoding operations. Specified 'exactly' or with 'marginals'

Non-increasing
Non-decreasing
Non-monotonic

If $A(i,j,...)$ is attribute value for selecting atoms up to $(i,j,...)$,
 $A(i,j,...) = A_{ref} \cdot f(i,j,...)$

If A_{ref} is default attribute value for dropping the component
 $A_{ref} = A_{ref} \cdot f_{ref}$

Ex. $A(2,1) = A_{ref} \cdot f(2,1)$

SCISM 24

SCISM: Transcoding Operation: Meta-data & Constraints

- Transcoder gets all the information it needs to decide which layers to drop from:
 - Meta-data in input Parcel header
 - Capabilities & Preferences Specifications
- Transcoder decides layer drops based on 'numbers' – need fast optimization algorithm

SCISM 29

SCISM: Transcoding Operation: Tasks

- Transcoder acts on a parcel only if constraint measures involve media attributes.
 - Optimization Decisions:
 - Chooses minimal layer drops meeting limit constraints.
 - Within the space satisfying limit constraints, optimizes the optimization measure if any.
 - Maintains Component Dependencies.
 - Maintains layer drop consistency for Tiers in Component Consistency List.
 - Best effort if not all constraints can be satisfied.
 - Monotone type flag aids optimization.
 - Bit-stream manipulation:
 - Drops atoms, repacks and updates TOCs appropriately

SCISM 30

Structured Content-Independent Scalable Meta-formats (SCISM)

Debargha Mukherjee, Amir Said
Presented by: Giordano Beretta

Compression and Multimedia Technologies Project
Imaging Technology Department, ISL
Hewlett Packard Laboratories

Outline

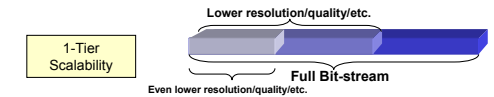
- Introduction
- Scalable Bit-streams Review
- Universal Meta-Format: SCISM
- Capabilities & Preferences Specifications
- Transcoding Operation
- Example Scenarios
- Conclusion

Introduction: Internet is Heterogeneous

- Diversity over the Internet is here to stay
 - Variety of Bandwidths
 - Wired: 56K/DSL/T1/T3/OC-X; Wireless: 2.5G/3G/4G.
 - Variety of Display Resolutions, Graphics Hardware
 - PDA/Notebook/Desktops/Visualization-Workstations
 - Variety of Processing Powers.
- The vision – 'Seamless ease of Media Use'
 - Won't materialize without adequate focus on media content adaptation.

Introduction: Media Adaptation Approaches

- Maintain multiple-versions of media:
 - Drawback: Storage & Bandwidth wastage
- Scalable bit-streams promise a better solution:
 - Smaller subsets of the whole bit-stream produce representations at lower quality, resolution, etc.
 - Can potentially eliminate all redundancies.
 - Simple transcoding from higher to lower version.



Introduction: Types of Scalability

- Known types of scalability:
 - SNR – Applies to most types of media
 - Resolution – Image, Video, 3D media
 - Temporal – Video, Image sequences
 - Interactivity – New interactive media
- New types of scalability will evolve with new media – 3D, 4D, Aromas etc.
- Scalability types may be combined

Introduction: Scalable Media Communication

- Mid-stream transcoders needed to reap the benefits of a scalable bit-stream.
 - Transcoders generate several versions from the same parent scalable bit-stream.
- Transcoders can be included in:
 - *Media servers* – media originates from here
 - *Mid-stream transcoding/routing servers*, or
 - *Edge servers* – recipients connect directly to them
- Scalable bit-stream advantage
 - Transcoders simply truncate & rearrange bit-streams.

Introduction: Scalable Media Infrastructures

- Transcoders often part of a larger infrastructure
 - Media moves down the transcoding chain while capabilities & preferences move up.

The diagram illustrates a media infrastructure. On the left, a 'Media Creator' (represented by a camera icon) is connected to a 'Media Server' (represented by a server rack icon). The Media Server is connected to a cloud labeled 'Scalable media'. Inside this cloud, there are several 'Transcoder' icons. Below the cloud is a box for 'Distribution Infrastructure' and another for 'Capabilities and preferences'. On the right, 'Media Consumers' (represented by a laptop and a smartphone icon) are connected to the cloud. Arrows indicate the flow of media from the creator through the server and cloud to the consumers, while capabilities and preferences flow from the consumers back towards the server.

SCISM 7

Introduction: Obstacle in Adoption

- Transcoders not media-independent
 - They need to be specially designed:
 - for the structure of a specific scalable media-type.
 - to interpret client capabilities & preferences relevant to a specific media-type.
 - Different transcoding infrastructures needed for different types of media
 - Infrastructure **too expensive** to deploy.
 - Standards provide little security, time-consuming.
 - Standards limit flexibility.
 - Adoption of Scalable Media need universality:
 - Same infrastructure to work with all media types

SCISM 8

Introduction: Objectives of this work

- To develop a framework for universal scalability, leading to evolution of *media-type agnostic* delivery infrastructures.
- Enables seamless ease of multimedia use.
- Need to standardize:
 - The media-format & meta-data [not encoding]
 - The Capabilities & Preferences Specifications

The diagram shows a 'Media Originator' sending 'Scalable media' to a 'Transcoder'. The 'Transcoder' outputs 'Transcoded scalable media' to a 'Media Consumer'. Below the transcoder, there are two boxes: 'Universal Meta-format (SCISM)' and 'Universal Capabilities & Preferences Specifications'. Arrows indicate that these specifications are used in the transcoding process.

SCISM 9

Introduction: Specific Objectives

The diagram shows a 'Transcoder' block. An arrow labeled 'SCISM-compliant scalable media' enters from the left. An arrow labeled 'Universal meta-format compliant transcoded scalable media' exits to the right. Below the transcoder, there are two boxes: '1. SCISM' and '2. Universal Capabilities & Preferences Specifications'. Below these boxes, there are two boxes: 'Meta-data' and 'Format Compliant Scalable Media Data'. An arrow points from 'Meta-data' to 'Format Compliant Scalable Media Data'. Below these boxes, it says 'Can be either XML-based or bit-stream syntax-based'. To the right of the transcoder, there is a box labeled 'XML-based'.

SCISM 10

Scalable Bit-streams: Properties

- To create a meta-format we need to:
 - Understand scalable bit-streams.
 - Capture the common characteristics of a scalable bit-stream in a loosely defined *bit-stream format*.
 - Add appropriate *meta-data* in headers describing the specifics for a given media.

SCISM 11

Scalable Bit-streams: Nested Scalability

- Multi-Tier Scalability
 - Scalable bit-streams inherently contain a nested scalability structure
 - Standards JPEG2000, MPEG-X are no exception.

The diagram shows a 3D structure of nested scalability. It consists of three tiers: Tier 1, Tier 2, and Tier 3. Each tier has a 'TOC' (Table of Contents) and two layers: 'Layer 0' and 'Layer 1'. The layers are nested, with Layer 0 of a higher tier containing Layer 0 and Layer 1 of a lower tier. This illustrates how a single bit-stream can be accessed at different levels of detail.

SCISM 12