In multimedia and hypermedia systems, the importance of mobile links to a stationary infrastructure and of mobile data terminals is increasing. Current advances in hardware technology for hand-held terminals (for example, display size and resolution, processing capabilities, and power consumption) let multimedia systems be implemented in mobile environments. Nevertheless, to fully reach that goal while meeting the end user’s expectations will require consideration of the specific limitations and usage scenarios of mobile environments. Therefore, we need to know which multimedia components are appropriate for mobile applications and how these components are related to the usage scenarios.

**Understanding the challenges**

Human perception is always integral and complete; humans use all their available senses to communicate. Consequently, human perception is multimedia. Also, human visual perception has very large reception potential but small action potential; audio perception has medium reception potential and medium action potential; tactile perception has rather small reception potential and large action potential. Only optimal exploitation of all these human communication primitives and their different potentials will lead to optimized user-centered development of applications. Therefore, developing effective mobile multimedia systems involves not only the inclusion of more and more media types but also consideration for the different human perception abilities.

**Developing effective mobile multimedia systems involves not only the inclusion of more and more media types but also consideration for the different human perception abilities.**

In general, each application or communication has a specific communicative goal. For this goal to be reached in a multimedia environment, these factors must be considered:

- the goal itself,
- the sense abilities of the human observer,
- the information to be coded,
- the application area with its metaphors and application-specific requirements, and
- the presentation and interaction devices and tools (multimedia appliances).

Furthermore, the development of multimedia systems involves two layers: the use of different media types and media integration, and the ability to handle different kind of media representation and levels of abstraction. For example, a cellular phone and a television set are two multimedia output devices; however, the information or content to be communicated to the end user must be represented by different media types—that is, audio for the cellular phone and video for the television.

**Multimedia information representation**

The integration of the various media types and the necessity to take into account communicative goals leads to the consideration of different levels of representation.

Each media type can be represented at different levels, starting from the optical or analog electrical representation, through the digital and feature representation, up to a symbolic representation, which is the most abstract level. Generally, the various representation levels are

- the transformation level, which deals with the spatial and time discrete media signals and their quantization;
- the feature level, which includes all media primitives and objects;
• the symbolic level, which is problem-oriented; and
• the presentation level, which generates the necessary media information output to assist humans with media-information perception.

Often, different representation levels are chosen to achieve different levels of quality or functionality. For instance, if the graphical media information is presented as bitmap images (digital pixel representation), the user can work with it in different ways than if it is represented as higher-level primitives (geometry and feature representation).

In general, higher levels of abstraction lead to reduced data rates and bandwidth. For example, a visualized computer animation (rendered images) has a data rate comparable with video, whereas the representation of the same animation in the geometry and feature domain using scripts, geometric objects, and properties achieves a highly reduced data rate without losing quality.

In addition to the integration of the media types, which is an essential feature of a mobile multimedia environment, the user-oriented development of mobile multimedia systems must take into account
• open, platform-independent systems and the integrated use of various delivery and communication channels;
• the improvement of application usability and communication; and
• media-quality settings and adaptation to communicative goals.

These aspects will likely determine the acceptance and usefulness of multimedia systems. Their consideration should direct system designs toward improved use of human-sensing potential and to an optimal design in accordance with application and user requirements, leading from a technological point of view to a user-oriented development.

An ideal mobile multimedia environment allows the representation of all media types at all different representation levels, as well as the conversion between the representation levels and communication primitives. Such an environment supports the conversion of a symbolic representation of media information to a visual, audio, or tactile representation, depending on the communicative goal and implementation restrictions. Most current R&D activity focuses on the conversion between different representation levels—for example, from analog to digital or from a feature representation to a digital image representation. Conversion between the different communicative primitives is much more difficult; it includes processes that are ill-defined, such as image and speech comprehension. Nonetheless, this kind of conversion in mobile multimedia systems will be important for user- and application-oriented development.

**Handling 2D and 3D Graphical Data**

Examining how 2D images and 3D graphics are processed will show us how information representation influences mobile multimedia applications.

Digital still and moving images are of a fixed size; they need specific compression for data reduction; and they have only limited integration capacity (that is, the ability to be included with other media types). 2D and 3D graphics and animation, which are the abstract basis for the generation of still and moving images, are more compact, so they do not need specific compression. Furthermore, they are based on objects, not pixels, and therefore offer high-level integration and lossless scalability. So, media information at the feature-representation level is suitable for generating mobile multimedia applications with high interactivity.

Using 2D and 3D graphics and animation in mobile multimedia applications allows value-added applications or services for such environments, because the obvious advantages of vector graphics over raster graphics can be fully exploited. Nevertheless, the 2D and 3D graphics formats used in mobile applications must be carefully chosen and adapted to the restrictions of the mobile environments. These restrictions comprise the low bandwidth of wireless narrowband wide-area networks, such as DSM and DCS-1880, and the limited resources of mobile hardware in comparison to stationary systems.

Using a 3D representation for certain objects—for example, objects of interest or the entire scene—adds realism to the presentation. Furthermore, such 3D descriptions are the basis for realistic animation scenes. The description of 3D vector graphics scenes requires different file formats from the description of 2D raster graphics.

The integrated presentation of 2D and 3D graphics and animation customized for mobile multimedia services requires a method for simultaneously handling and displaying these media types and for using 2D data, 3D data, animation, or a combination in one displayed scene. This method requires a plug-in implementation that also handles communication with the servers; for example, downloading additional data for integration in the displayed scene. Furthermore, it requires support for a number of user-interaction mechanisms, tailored to the data and the services presented in the implementation.

**Creating Graphical Content for Mobile Multimedia Environments**

The design of 3D presentation scenes for mobile multimedia applications must take into account the complexity of the scenes. This involves not only the complexity of the transmission of data over a low-bit-rate channel but also the complexity of the presentation of the 3D scene on the mobile client. These considerations always lead to a trade-off between scene-presentation quality and complexity.
An important aspect of scene complexity is the relation of file size to the scene’s information content. For instance, in a graphics environment, objects representing the same information can be described in different ways, which eventually leads to different file sizes. Therefore, the “smallest” way of describing graphics objects must always be used. That is, using geometric primitives, whenever they are applicable, instead of polygonal description will produce smaller data files. Moreover, when a scene description uses the same graphics object several times, multiple definitions of the object should be avoided. Also, the amount of data to be transferred over the mobile link to the client should be reduced by using local resources as much as possible—for example, using texture scripts and audio files from the local hard disk instead of transmitting them over the mobile network.

Regarding the complexity of the presentation of the 3D scene in a mobile environment, the main consideration is the application of visual effects (such as light, texture, background, and fog). The purpose of such effects is to make a scene more realistic. However, these enhancements place an additional load on the mobile client’s computational resources. So, the number of textures, lights, and so on used in a scene description must be kept at a minimum. Furthermore, the number of points and faces, as well as the key-value pairs for key-frame animations, should be decreased as much as possible.

Apart from graphical-content presentation, a number of different tools automatically create content for mobile services, optimizing that content for mobile multimedia environments or adapting it to the limitations of the environments. These tools generate 3D-scene content and dynamic behavior for the content according to specific requests. The tools use different templates, each containing a graphical description of a 3D object that can be configured to the requirements of the requests: that is, the specification of the objects’ geometry (for example, colors or textures applied to the objects), the object’s location in the 3D space, and so on. Several of these objects can be combined in one layer and written to an output stream using the graphics file format. Moreover, each 3D object can be configured to handle different user interactions, perform different animations, or both.

Generally animation sequences and 3D scenes generated by such tools encode the data using an ASCII representation, which is unsuitable for transmission over mobile links. Enhancing the added value by specifying more complex scenes without increasing the amount of data to be transmitted requires data compression. This functionality is usually integrated in the content-creation chain. The data-compression function removes redundant data (for example, white spaces and redundant digits) from the encoded file. It also produces highly compressed vector graphics and animation data types by using a set of predefined dictionaries that are adapted to the content types.