H2O 1.0

Product Requirement Document

Document Number:	
Author:	Simon Trudelle
Creation Date:	November 11, 2000
Last Updated :	April 14, 2002
Version:	0.8
Location:	OTV_H2O_1.0_PRD_8.doc

OpenTV Inc.
401 East Middlefield Road
Mountain View, CA 94043

OpenTV Confidential – Do Not Distribute

TABLE OF CONTENTS

TABLE AND FIGURES

REVISION HISTORY

Version	Author	Date	Description
0.01	Brad Fuller, Mic Michelsen, Greg Nawrocki	10-Nov-00	Initial Version
0.02	Brad Fuller	?	Clean up
0.03	Brad Fuller	09-Dec-00	Added Functional, Architectural and Performance Requirements
0.04	Brad Fuller, Mic Michelsen, Greg Nawrocki	12-Dec-00	Added updates and Documentation, Training, Constraints, etc.
0.05	Brad Fuller, Mic Michelsen, Greg Nawrocki	14-Dec-00	Grammar
0.06	Brad Fuller	22-Dec-00	Updated with input from Alain Delpuch, Danny Su
0.07	Brad Fuller	27-Dec-00	Updated with input from Alain Delpuch and Pierre Willard
0.08	Simon Trudelle	Apr-02	Updated for 1.0 version

1 Definitions, Acronyms, and Abbreviations

Words used in this document to define requirements are significant and are defined below:

"MUST"	Means that the item is an absolute requirement of the specification. Adjective: "REQUIRED"
"MUST NOT"	Means that the item is an absolute prohibition of the specification.
"SHOULD"	Means that there may exist valid reasons in particular circumstances to ignore this item, but the full implications should be understood and the case carefully weighed before choosing a different course. Adjective: "RECOMMENDED"
"SHOULD NOT"	Means that there may exist valid reasons in particular circumstances when the listed behavior is acceptable or even useful, but the full implications should be understood and the case carefully weighed before implementing any behavior described with this label.
"MAY"	Means that the item is truly optional. Adjective: "OPTIONAL"

2 Purpose

The purpose of this document is to describe the product requirements for a product - ***OPENTV H2O*** - that enables content providers and e-commerce operators to use their current HTML and Javascript based Internet content development tools, knowledge base, and application server infrastructure to create OpenTV interactive television (iTV) applications.

This updated document supersedes all the previous versions.

3 Scope

The scope of this document contains the entire high-level product definition of the *OPENTV H2O* product. It is an update to the original document and now includes references to other documents (the Detailed Requirement Documents and FFS written during the R&D phase) that provide the detailed list of supported features.

The **Phase 1 requirement section at the end** defines the requirements for the implementation of an e-commerce managed service on the BskyB platform.

This document also refers extensively to other OpenTV products that are mandatory components of the complete H2O solution.

- OpenTV Streamer (including the Channel Server feature)
- OpenTV Gateway
- OpenTV Core (including EN and EN2)
- OpenTV HTML Tools (TBD)

4 Reference Documentation

Title	Author	Ref. Number	Revision	Creation Date
OpenTV Web 1.0 User Guide	OpenTV	4-100-20-00		Jul 98
OpenTV Technical Council Presentation	Technical Council	-	V2.0	24 Oct 00
H2O Framework Component	Alain Delpuch	-	V0.6	7 Dec 00
H2O PRD (initial draft)	Brad Fuller	-	V0.7	27 Dec 00
H2O MRD	Simon Trudelle	-	V0.2	3 Apr 02
H2O Detailed Requirements	Emmanuel Barbier, Simon Trudelle	-	V0.3	15 Apr 02
H2O HTML2O FFS	Project Team	-		
H2O HTML2O DOM FFS	Project Team	-		
H2O JS2O FFS	Project Team	-		
H2O HTML-TV FFS	Project Team	-		
H2O Proxy FFS	Project Team	-		
H2O Carousel Manager FFS	Project Team	-		
H2O Set-top box integration FFS	Project Team	-		

Table 1 - Reference Documentation

5 Executive Summary

The *OPENTV H2O* product is a product answer to help bridge the gap between thin set-top boxes running OpenTV middleware technologies and widely deployed Internet technologies.

The product aims to leverage the current technology base, development knowledge, and application server infrastructure of the Internet while retaining the benefits of the OpenTV technology. The result is a solution to enable content providers to access the installed and future base of OpenTV-enabled receivers with a minimum change to their existing technology infrastructure.

The *OPENTV H2O* product is a client and server solution that relies on Device Mosaic on the server side and leverages the features of existing Network Solutions products on the client side. The result from the client-side component is an O-code application that conforms to the OpenTV Application Framework Architecture. The server-side component is an HTTP proxy that is an intermediary step between a content provider's application server and OpenTV's network infrastructure (point to point and broadcast tools).

The *OPENTV H2O* product general features are:

- HTML 4.0 /JS 1.4 solution for entry-level set-top boxes
- HTML/JS support for TV-Centric (content authored for television) applications
- Low upfront investment solution, relying on a scalable server architecture
- Flexible solution for monetizing content, allowing a 'push or pull' content distribution model (broadcast and point-to-point)
- Fully scalable solution, with high availability

The *OPENTV H2O* product can be seen as a middleware extension for entry-level set-top boxes, with a server-side component that scales up with demand for two-way interactive services.

The product could be marketed either as a solution for operating a managed-service HTML conversion solution or as a software product.

6 Product Concept

6.1 Positioning

The H2O product is targeted as a new HTML/JS-based solution for deploying applications on thin set-top boxes. While H2O supports the latest Web standards, some functionality will not be supported due to architectural constraints. As well, since HTML/JS is not a solution originally designed for set-top box middleware and hardware, new features are added to allow TV-centric authoring.

6.2 Architecture

Although not a product requirement, *OPENTV H2O* is by default made up of two components: server and client component:

- The **server** side of *OPENTV H2O* is named *H2Os.* It is an HTTP proxy.
- The **client** side of *OPENTV H2O* is named *H2Oc.* It is an OpenTV O-Code application

Most of the power of *OPENTV H2O* resides on the server side component: *H2Os. H2Os* is essentially an HTTP proxy server that utilizes OpenTV's communication infrastructure. As an example, a typical transaction between *H2Oc* and *H2Os* would be the following:

H2Oc would request a particular asset (a page, jpg image, etc.)

If the page is in the broadcast path or resident, the page would be displayed accordingly.
If not, it would request H2Os for the page. H2Os would fetch, compile and return the page to H2Oc.

H2Oc displays page

(Note that *H2Os* might be used to convert broadcast pages. In this case, requests are doming from the OpenTV Streamer Channel Server.)

Note that some of the assets for the requested page may already be resident or come from either the broadcast or online path.

H2O also provides support for new HTML-TV *OPENTV H2O* tags to handle the TV Librairies features available within the OpenTV Middleware (Audio/Video control, Channel control, OSD controls, Triggers).

6.3 Tools to create OPENTV H2O pages, logic and assets

While OpenTV HTML tools are not included in the scope of this document, let's mention possible options:

- An HTML/JS checker that checks the compliancy of HTML pages for H2O
- H2O plug-ins to favorite Web design tools (such as Macromedia Dreamweaver)
- COM object libraries (from the SDK) for converting assets (images, fonts)
- JS debugger

6.4 Licensing

The H2O product has to be marketed as one product with multiple components and each component needs to have license protection to avoid illegal copying and implementation.

The H2O product is similar to other server-side OpenTV products and has to be available as a development tool and a production system, including a Test & Commissioning version.

6.5 Parties and systems interacting with the product

- H2O Platform Operator
  The H2O Platform Operator is the commercial operator of the H2O product. Typically, it is the Network Operator but it can be a third-party (OpenTV) providing a managed-service solution. The H2O Platform Operator has to be able to track the traffic on the platform to bill its customers
- Network Operator
  The Network Operator is the operator of the digital TV network. It needs to provide support to guarantee that the H2O browser is validated on all the deployed set-top boxes.
- System Hosting Operator
  The System Hosting Operator is the operator that will run the hardware, software and network infrastructure required to operate H2O-based services. It can be a specialized third-party hosting company hired by the H2O Platform Operator or the function can be handled by the H2O Platform Operator or the Content Providers directly
- Content Providers (broadcasters, publishers, retailers, advertisers)
  The Content Providers operate the applications (or services) authored in HTML/JS, providing TV-centric and H2O-compliant content. They also determine which pages should be included in the broadcast carousel, and which pages need to be accessed online.
- Content Developers (application developers)
  The Content Developers are graphic designers, UI specialists and engineers that will implement TV-centric applications authored in HTML/JS using Web tools and technologies and iTV HTML tools as needed
- OpenTV Account
  OpenTV Account provides viewer registration/identification and transaction logging functions that Network and/or H2O Platform Operators may want to implement in parallel with the deployment of H2O
- OpenTV Streamer
  OpenTV Streamer and the Channel Server extension handle the fetching of content and the carousel building functions

The product is likely to be bought by a party (the H2O Platform Operator), operated by a second party (the System Hosting Operator) and will be used to process content delivered by external content providers (a Retailer like eBay). Moreover, the resulting applications may run on multiple networks.

Consequently, additional requirements have been added into this document to specify the needs of each of the potential users of the product.

7 Product Name

OPENTV H2O is the product name. It stands for ‘HTML-to-OpenTV (O-code)’.

8 Target Customers/Market

The product will be used by customers for implementing multiple types of iTV applications on a large scale. However, the key target market is t-business applications – i.e. applications making intensive use of two-way interactivity through the return channel to monetize content.

Target customers for production systems are:

- Content Providers (Web e-commerce and service operators with the potential of launching services across multiple platforms worldwide (around 10 prospects worldwide: eBay, Amazon, Ticketmaster, Expedia, …)
- Network Operators or H2O Platform Operators wanting to process HTML/JS content authored internally, third-parties developers or Content Providers

ITV application developers are the target market for the developer's version.

9 Functional Requirements

9.1 Overview

This section provides an overview of the functional requirements identified for the H2O Product. The detailed list of the retained requirements is covered in the following sections.

The requirements are numbered in order to ease their tracking in the product development process. Each functional area is sub-divided in sub-areas, and each requirement has a requirement number using the following scheme:

<functional area>, <sub-area>, <requirement #>

The requirements are organized in the following groups:

1. Types of applications
2. General requirements
3. Specific functional requirements
4. System operations and administration requirements
5. Performance requirements
6. Compatibility requirements
7. Installation requirements
8. Packaging requirements

9.2 Types of application requirement – 1.1

Background

Basically, 5 major types of iTV applications can be implemented. H2O is specifically aimed at implementing the first three types of applications.

Requirements

Req #	Type	Exemple	H2O requirement
1.1.01	Virtual Channel applications (do not use video at all) requiring some application logic	t-commerce, magazines	MUST

1.1.02	Enhanced TV applications, more or less synchronized with the current video, requiring some application logic	Broadcaster's eTV content, eTV advertising	MUST
1.1.03	Virtual Channel Portal Applications that can select different streams in the current channel or switch to a different channel	TV Mosaic application linking to other TV channels / virtual channels	MUST
1.1.04	Communications applications (Chat, Mail, SMS, …) that require specific communication protocols, some application logic and virtual keyboard	Chat application module integrated with other content	MUST
1.1.05	Resource intensive applications requiring access to low-end functions and/or application logic	Games	SHOULD
1.1.06	Applications that can access all the SI and CA information (often embedded)	EPG, pay-per-view	MAY

9.3 ***General requirements – 2.1***

2.1.01 *OPENTV H2O* MUST convert a subset of Web content (such as HTML pages, JPEG pictures, business logic in the form of ECMAScript, etc.) into OpenTV resources and O-code logic to be presented and executed on OpenTV-enabled STBs.

2.1.02 *OPENTV H2O* MUST support the broadcast of dynamic content and allow seamless access to online content over the return channel.

2.1.03 *OPENTV H2O* MUST interoperate with the Cougar/1.2 http implementation, OpenTV Gateway 1.3 (lhttp) and the EN2/OpenTV Core http extension

2.1.04 *OPENTV H2O* MUST support uncompromising secure transactions required by e-commerce servers for retailers, banks, etc. These requirements are SSL v2/v3

2.1.05 *OPENTV H2O* MUST support a large number (thousands) of concurrent online connections and implement a fully scalable server architecture (tens of servers)

2.1.06 *OPENTV H2O* MUST make efficient use of broadcast and online bandwidth for cost-effective implementations, including the sharing of the application browser code among many services (on the same transponder)

2.1.07 *OPENTV H2O* MUST be similar or exceed OpenTV's current television experience; i.e. OPENTV H2O MUST NOT compromise on the current user quality TV experience that viewers are accustomed to.
Functionality, feature and presentation capability (such as positioning, hiding, rendering speed of gadgets, ease of navigation) MUST NOT be less quality than the current OpenTV Publisher 1.3 product.

2.1.08 *OPENTV H2O* MUST include HTML tags for implementing specific TV-centric features provided by the OpenTV Middleware. These tags SHOULD be normalized by industry Standards Forums in the future and SHOULD be interoperable with competing iTV

platforms.
[The expectation is strategic: it would then be possible for Static to author content for multiple, competing platforms supporting HTML/JS for television]

2.1.09 Application developers MUST be able to author content for OPENTV H2O without using other existing OpenTV specific tools (SDK, Frame).

2.1.10 Application developers MUST be able to create applications delivered in MUX file format (and thus do not require OpenTV Streamer)

2.1.11 *OPENTV H2O* SHOULD NOT support specific tags implemented by competitors (Liberate and others) – note: we may support these tags if it is judged necessary to gain market acceptance

2.1.12 *OPENTV H2O* SHOULD NOT support standard plug-ins (like Macromedia Flash and other) – note: technical limitations make it too complex

2.1.13 ITV developers MUST develop their services specifically for television and H2O. We assume that iTV developers will "re-publish" existing or create new services specifically for OPENTV H2O

2.1.14 Content developer SHOULD be able to author content using popular tools (such as PC Web page editing tools)

2.1.15 OPENTV H2O SHOULD be easily interfaced to new OpenTV products (i.e. when a new middleware/server product arrives, the H2O product should not have to be modified)

2.1.16 Third-party tool developers SHOULD be easily accommodated so that they can easily develop new authoring tools for H2O

2.1.17 The OPENTV H2O tag specification SHOULD be published in such a way for the possibility of it becoming a standard in the iTV e-commerce community

2.1.18 Content authored for Liberate (HTML 3.2/ JS 1.2) and PowerTV MAY be executed on *OPENTV H2O* but it is understood that most pages may have to be re-authored

9.4 Functional Requirements - 3

9.4.1 OpenTV H2O Client: H2Oc

9.4.1.1 H2Oc general requirements – 3.1

3.1.01 *H2Oc* MUST provide functionality for the display and use of assets (e.g. positioning of images, use of fonts, CLUTs, etc.)

3.1.02 *H2Oc* MUST provide functionality for the execution of logic and provide the capability to merge assets or logic from either the broadcast (in-band) or the online (out-of-band) path.

3.1.03 *H2Oc* MUST be a compliant O-Code application.

3.1.04 *H2Oc* MUST be a compliant OpenTV Network Solutions client component. It MUST utilize Network Solutions libraries such as the OpenTV Streamer (including bHTTP) and Gateway library or the carousel load library.

3.1.05 *H2Oc* MUST run on current 1.0, Cougar/1.2, and EN and EN2 middleware.

3.1.06 *H2Oc* browser code MUST be shared across multiple applications on the same transponder, allowing also a service switch without reloading the browser

3.1.07 *H2Oc* browser code MUST be implemented in such a way that it can be flashed permanently in the set-top box memory

3.1.08 *H2Oc* browser MUST provide a flexible way to implement a configurable splash screen when the browser is loaded. The splash screen MUST be very light and contain only a limited number of animation features (display text like 'launching….' With an animation – 'clock ticking…')

3.1.09 *H2Oc* MUST provide the option to:
- Display required assets upon reception (as assets are loaded) or
- Wait for all assets to be loaded and display the page in one shot.

3.1.10 *H2Oc* MUST be able to share assets, and other data, efficiently across *H2Oc* pages by managing a local cache for the length of the session.

3.1.11 *H2Oc* MUST use the OpenTV Streamer Carousel library to dynamically manage broadcast resources (i.e. creating new resource, deleting resources, changing resources).

9.4.1.2 H2Oc logic – 3.2

3.2.01 *H2Oc* MUST provide the functionality to execute logic that is compatible with a subset of Javascript 1.4 (or higher).

The OpenTV Javascript Subset is converted to O-code logic at the *H2Os* server and sent to *H2Oc* for execution as a complete O-code application.
The reference document defines this subset.

3.2.02 H20 MUST support dynamic O-Code plug-ins or new gadgets that can be created using the SDK.
{Elements should come as O-Code DLL (or equivalent) so to be able to substitute them for different look and feel. Another idea is that we'd broadcast only the one that are really needed.}

9.4.1.3 H2Oc Presentation/User Interface – 3.3

3.3.01 *H2Oc* MUST be able to support the real-time update of 'pushed' broadcast content. It MUST keep displayed content consistent during updates of the content.

3.3.02 *H2Oc* MUST support the various video scaling extensions available on EN2, 1.2 and Core.

3.3.03 *H2Oc* MUST provide a configurable user interface (OSD messages/ 'lock' icons) for informing the viewers about the secured status of the online connection

3.3.04 *H2Oc* MUST support the specific HTML-TV requirements defined in the H2O Detailed Specifications Document.

9.4.2 *H2O Server:* H2Os

9.4.2.1 Architecture requirements – 3.4

3.4.01 *H2Os* MUST execute on the Windows 2000 platform

3.4.02 *H2Os* MUST scale to up to 255 servers

3.4.03 *H2Os* MUST be implemented as a standard COM object to provide complete scalability. Any third party technology MAY be used to provide the required scalability (upon agreement by Product Management that the cost of such technology is acceptable)

3.4.04 *H2Os* MUST re-distribute the load seamlessly when an *H2Os* process/machine fails.

3.4.05 *H2Os* MUST make sure that Web content outside the 'walled garden' cannot exploit security breaches in order to hijack the set-top box (for instance to provide Pay TV for free).

3.8.01 *H2Os* MUST be designed and implemented so that the Javascript compiler can be updated easily, without impacting other components of the system

9.4.2.2 Converting content to OPENTV H2O Content – 3.5

3.5.01 *H2Os* MUST parse TV-centric content to produce *OPENTV H2O* pages (which is essentially OpenTV resources and O-Code.) TV-centric content is content that has been created specifically for iTV using Web technologies like HTML 4.0, Javascript 1.4 and a variety of assets listed below.

3.5.02 When *H2Os* encounters improper syntax (be it HTML, JavaScript, or assets) it MUST produce an appropriate error process (log/display error code) and reject the complete HTML page. For online access, the viewer must receive a configurable error page indicating that his page request failed.

3.5.03 When *H2Os* encounters unknown document types it MUST pass-thru the document type unchanged and it MUST produce an appropriate error process (log with URL and error code). For online access, the viewer must receive a configurable error page indicating that his page request failed.

9.4.2.2.1 H2Oc assets and DOM features supported – 3.6

3.6.01 HTML 4.0.1 with some limitations expressed in H2O HTML2O FFS

3.6.02 CSS with some limitations expressed in H2O HTML2O FFS

3.6.03 Javascript 1.4 with some limitations expressed in H2O JS2O FFS

3.6.04 Jpeg, gif, bmp image formats

3.6.05 OpenTV HTML TV extensions to access TV-centric features, as expressed in H2O HTML-TV FFS

3.6.06 OpenTV asset types as expressed in H2O HTML-TV FFS

9.4.2.2.2 HTML Parser – 3.7

3.7.01 *H2Os* MUST support the standard HTML 4.0.1 tags defined in the H2O Detailed Requirement Document

3.7.02 Additional HTML tags are required to extend HTML for interactive TV. The HTML parser MUST be designed so that adding tags in the future will be easy and cost-effective.

3.7.03 The HTML parser MUST support major latin-based languages and SHOULD support all major international languages – Arabic, Hebrew, Asian (support for UNICODE characters)

9.4.2.2.3 Javascript Compiler to O-Code – 3.8

3.8.02 *H2Os* MUST compile a subset of ECMAScript 262 (Javascript 1.4) to OpenTV O-Code. The list of the required functions can be found in the document H2O Detailed Requirements document [x]

10 System operations and administration requirements

10.1 Monitoring and remote management – 4.1

4.1.01 *H2Os* MUST provide a Web User Interface for remote management. The 'look & feel' of the interface MUST be compatible with other new generation OpenTV products

4.1.02 *H2Os* MUST use standard monitoring techniques, based on SNMP, so that local and remote monitoring of functions are built-in

4.1.03 *H2Os* SHOULD use standard monitoring techniques compatible with the OpenTV Administration/Monitoring System Guidelines (under definition for PS implemention)

4.1.04 *H2Os* MUST provide monitoring of basic functions (i.e. running/not running, processes complete, compiled modules/pages, major problems, etc.)

10.2 H2Os usage logging and reporting – 4.2

4.2.01 *H2Os* MUST log all the online HTTP asset requests [NOTE: need to double check what Account could do already], making a clear distinction between assets served from the proxy cache and the assets fetched from the Content Provider and converted by H2Os

The log MUST track the following data fields:

- Date/Time stamp
- Asset type extension (htm, jpg, gif, …)
- Asset size in Kb
- Source (E=external, P=proxy)
- Viewer identifier (tbd)

4.2.02 H2Os MUST log all the online sessions, tracking the following data fields:
- Date/Time stamp (start session)
- Session duration in seconds
- Viewer identifier (tbd)

4.2.03 H2Os MUST build log files that can be accessed with a standard tool (tbd) to produce usage reports.

4.2.04 H2Os MUST provide functionality to determine the time window of the online log and to manage exporting/archiving of the log (the archived log MUST be easily accessible to extract usage reports)

4.2.04 H2Os MUST ensure that the log file can be backed up without having to shut down the system, that the log file cannot be tempered with – the log file has a commercial value as it can be used to do billing

4.2.05 H2Os MUST provide a reporting module to produce user-defined reports by accessing the log file or a database created with the log file

10.3 Auditing – 4.3

4.3.01 The log data MUST be accessible from a remote location by a customer, with the appropriate security level to look at the log reports

10.4 Security – 4.4

4.4.01 The access to the log file MUST be controlled with three security levels:
- basic user status: generate designated reports

- master user status: generate full reports, create users
- administrator status: administer / archive reports

4.4.02 Only authorized users MUST be able to update (archive only) the log file.

11 Performance Requirements

11.1.1 Performance

11.1.1.1 H2Oc – 5.1

5.1.01 Graphic rendering performance of an average H2O page MUST be completed in about 0.5 second on a TP3 Pace decoder box.

5.1.02 {Performance requirement for JS scripting execution must be specified here – needs to be added}

11.1.1.2 H2Os – 5.2

5.2.01 Converting an HTML/JS page (including images) MUST be on average 0.1 second per page (average weight of around 100 M cycles – needs to be validated) on a 1 Ghz CPU machine.

5.2.02 H2Os MUST add an overhead in total of no more than 0.25 seconds elapsed time to an average page access

5.2.03 H2Os MAY use special hardware and/or software to optimize the transcoding performance if the cost of this extra power is economically justified (i.e. cheaper than adding more general-purpose servers)

5.2.04 Billing reports MUST take less than 1 hour to generate by 100 million transcoded pages (needs to be validated).

11.1.2 Sizing – 5.3

5.3.01 H2Os MUST support up to 250,000 concurrent users (to be validated) and not see a major degradation in performance when handling such a peak load

5.3.02 H2Os MUST support up to 10 million viewers on one or many networks

5.3.03 H2Os MUST support up to 2500 (to be validated) page conversions a second (peak) – using up to 255 servers

5.3.04 H2Os MUST support up to concurrent 200 applications (OpenTV Account requirement as well)

5.3.05 H2Os MUST support up to 50 Content Providers (OpenTV Account requirement as well)

5.3.06 H2Os MUST store in a cache as much transcoded content as required by the Content Provider to optimize performance

11.1.3 Stability – 5.4

5.4.01 H2Os servers MUST not require to be re-booted more than twice a year

5.4.02 It MUST be possible to do a start/stop of any H2Os servers at anytime without impacting the rest of the system (current H2Os servers running and administration tool)

12 Compatibility requirements

12.1.1 Backward Compatibility – 6.1

6.1.01 No backward compatibility requirements other than the ones stated in other sections.

12.1.2 Compatibility and inter-operability with resident Applications – 6.2

6.2.01 No compatibility requirements different than with other applications.

12.1.3 Compatibility with other OpenTV Products – 6.3

6.3.01 *H2Oc* MUST be compliant with the OpenTV Framework Application Architecture.

6.3.02 *H2Oc* MUST be compatible with the OpenTV Gateway 1.3 HTTP functionality.

6.3.03 *OPENTV H2O* MUST be compatible with OpenTV Streamer 2.1 and the Channel Server extension.

6.3.04 *OPENTV H2O* MUST be compatible with OpenTV Account 1.2

12.1.4 User Interface/Interaction Requirements – 6.4

6.4.01 The System Installation and Administration screens MUST be compliant with the OpenTV UI standards for product (including the OpenTV Account Administration Tool and OpenTV Streamer Administration Tool – Channel Server)

13 Installation Requirements – 7.1

7.1.01 H2O MUST be installable by system administrators / engineers (production version) and developers (developer's version) using the delivered documentation.

7.1.02 The product MUST allow the installation to be done on a standard desktop Windows machine (developer's version) as well as on multiple servers (production version).

7.1.03 The installation program MUST allow installation of multiple server components (transcoder, administration, billing modules, …) separately on different machines.

7.1.04 The installation process MUST be compatible with a direct distribution model (shrinkwrap license agreement and downloadable software) for the developer's version.

7.1.05 All Electronic versions of the documentation MUST be installed / not installed on the *H2Os* server during the installation process, at the user's choice.

7.1.06 The product MUST be installed/de-installed automatically using an InstallSheild program.

14 Packaging Requirements – 8.1

8.1.01 The product MUST be delivered on a CD-ROM with the appropriate product ID/license certificate and product documentation for installation, configuration, administration, application development and support information.

8.1.02 The same CD-ROM MUST be delivered for the two types of versions required:

- Production version
- Developer version

8.1.03 Additional tools for the developer's version SHOULD be delivered separately (Tools' product).

14.1 Artwork

The artwork MUST be compliant with OpenTV current packaging standards

The following artwork is required:

- CD Silkscreen artwork
- CD Jewel Case artwork
- Binder Cover for Docs
- Bitmap Splash Screen BMP for InstallShield
- HTML templates for Administration and Billing tool
- HTML templates for the Tutorial application
- Background video (no IP rights needed) for the Tutorial application

14.2 Versioning

The two most significant commercial version numbers are reserved for changes in functionality.

The first version will be 1.0.

14.3 Tutorials

The product MUST be delivered with ONE tutorial application that can be used for validating the system on a commercial network and to demonstrate the detailed functionalities supported by the product on a trade show or for customer presentations.

The Tutorial application MUST NOT need the use of additional specific Web tools (like an ASP server, databases, etc…) and SHOULD be made with static HTML/JS pages. It MUST make use of the background video.

A specific requirement document will outline the features of the Tutorial Application.

15 Channel Requirements

15.1 Direct Sales

An evaluation version of the product for Authoring/Testing MUST be available for important prospects. This version has to be identical to the regular developer's product.

15.2 Networks

Bandwidth Simulation tools have to be provided to allow Networks to decide whether or not it is worth considering embedding the H2Oc component.

15.3 Manufacturers

For manufacturers integrating the H2Oc component as embedded software, an extension to the HPK documentation has to be provided to support the implementation of H2Oc.

15.4 Chip Set Vendors

N/A

15.5 Integrators / Hosting Service providers

Capacity planning tools with related documentation have to be provided to allow third parties to determine the required hardware and network infrastructure for operating H2Os on a given network

16 Third-Party Technologies/Partners

The H2O product is reliant on a number of third party software products that are not part of the part and have to be purchased separately.

Company	Technology	License Terms
Microsoft	Microsoft Internet Information Server	To be acquired by the customer
Not specified	Proxy	To be acquired by the customer
Not specified	SNMP monitoring tool	To be acquired by the customer
Not specified	Report generating tool	To be acquired by the customer

Table 2 - Third-Party Technologies/Partners

17 Quality Requirements

H2O requires a high level of availability, similar to OpenTV Account. ITV Services need to operate on a 24*7 basis. Therefore, H2O should operate with 99.9999% availability, meaning with only 2 scheduled system downtimes a year of less than half an hour each.

Offline components will not require such high levels of availability.

The availability statistics of H2O that will be communicated to customers MUST be equal or greater than 99.98%.

18 Testing Requirements

18.1 Scope

All the functional requirements (including all the tags, features and functions) MUST be tested individually.

A number of complex test applications and scenarios (no more than 10), mixing the most commonly used functional features, MUST be developed by the QA team, validated by Product Management and executed.

All the non-functional features (scalability, performance, stability, ….) MUST be tested in a context relevant with the actual use of the product by customers, including interfacing with other related OpenTV products.

19 Documentation Requirements

All customer documentation MUST be professional documentation (copyright information, standard formatting and indexing, etc.)

19.1 Internal

A living database (HTML forms and search engine) MUST be available to document all the 'implementation best practices' used to develop H2O applications, including a category for 'work-arounds' used circumvent functional limitations or known bugs.

19.2 External

The following technical documents are required

- Getting Started
- H2Os Installation Guide
- H2Os Administration Guide
- H2Oc User Guide
- H2O Programming Guide

Future releases (updates and upgrades) of OPENTV H2O will require a migration guide to assist customers in the migration to the new release.

An additional guide is required for integrating H2Oc as an embedded component using the HPK.

19.2.1 Getting Started

This is a manual for system administrators, network engineers and application developers. It provides an overview of the system and its components. It provides the specific information for implementing the product within the target environment, including the relationships with other OpenTV products.

19.2.2 H2Os Installation Guide

This is a manual for system administrators and application developers that will install the H2O product. It must describe the installation and configuration options and the license activation process.

19.2.3 H2Os Administration Guide

This is a manual for system administrators and network engineers. It provides all the detailed information required to operate the product and generate usage/billing reports.

19.2.4 H2Oc User Guide

This is a manual for application developers. It provides mainly the list of supported tags and features, with examples tied to the Tutorial.

19.2.5 H2O Programming Guide

This manual is for application architects and advanced developers. It describes how to create complex H2O services and it offers guidelines for creating optimized HTML/JS pages for H2O.

This manual should have between 300 and 500 pages.

19.3 Printed

H2Os Installation Guide (except for the Developer's version – no documentation).

19.4 Electronic

Electronic versions of the printed version must accompany the CD(s) and be optionally installed on the server where *H2Os* is installed.

20 Training Requirements

20.1 Trainers

Trainers MUST have near-expert knowledge in the following:

- Installation and operations of H2Os (for Technical Support, Professional Services and eventually for 3rd party system integrators)
- Creating iTV services and applications in HTML/JS (for internal and external OpenTV application developers and Technical Support.)

20.2 Customers

Training requirements are for the following profiles:

- System Administrators
- Network Engineers

The System Administrators and Network Engineers need to know how to plan the implementation, do the installation and configuration and maintain OPENTV H2O. The also need to know how to operate the system and generate operations reports.

20.3 Third Parties

Training requirements are for the following profiles:

- Application Developers
- System Integrators

The Application Developers need to learn how to develop applications that are fully compliant with OPENTV H2O and are optimized for performance.

Systems Integrators are OpenTV Professional Services or third-parties that will do the integration of OpenTV H2O, including interfacing it with other systems (monitoring, reporting, etc.).

21 Technical Support Requirements

Support needs to cover the following areas:

- Operational
- Application development
- Technical Integration

21.1 1st Level

Technical Support MUST be the 1st level of technical support provided by OpenTV. It MUST be 24x7 support for OPENTV H2O.

21.2 2nd Level

2nd level of support will be customer and H2O focused, and offered during business hours on a continental basis (same time zone). This specific H2O Customer Support Team (Professional Services or Technical Support - TBD) will assist both system administrators and application developers in implementing solutions for OpenTV H2O. This team will assess the support requests and identify the specific issues that need to be addressed in priority by the Product Engineering group responsible for the H2O product.

21.3 Online

Dedicated e-mail box managed by the 2nd level support team.

22 Constraints

22.1 Dependencies

22.1.1 OpenTV Streamer / Channel Server

OpenTV Streamer has to be extended to implement the Channel Server solution for managing the fetching and building of the broadcast carousel.

22.1.2 OpenTV Account

OPENTV H2O should be implemented with OpenTV Account to provide an efficient way to manage personal viewer information (login/password) within H2O-based applications.

22.2 Time Scope

In order to allow Sales/BD to be confident in pushing the product to customers and to maintain a close relationship between commercial results and product investments, the following calendar MUST be followed:

T0 Project approval

T+2 Pre-Alpha

T+4 Alpha

T+6 Beta

T+8 FCS

To support the sale of the product at the beginning, an intermediate pre-alpha release is needed to provide a packaged development environment for testing HTML/JS content authored for H2O. This intermediate release is aimed at individual developers testing applications on a target decoder. The online access does not have to scale but MUST be fully functional. It MAY use implementation techniques that will be different in the final product, as long as the content authored for this pre-alpha release stays compatible.

22.3 Technical constraints

H2Os performance objectives and scalability MUST be guaranteed. The project MUST be reassessed if these objectives cannot be met.

22.4 Customer or Market constraints

Sufficient compatibility with existing Web technologies used by content providers and commerce operators is necessary.

The product MUST be released in 2002 (see MRD).

23 Legal Issues

No unusual legal or licensing issues expected.

23.1 Licensing

23.2 Copyright

End of document

24 Specific Phase 1 requirements

Phase 1 consist of the implementation of t-commerce applications as a managed service on the BskyB platform.

From the list of requirements expressed in this document, the following requirements MUST NOT be implemented for Phase 1 and MUST be addressed in a later phase.