Hyper-G Client/Server Protocol (HG-CSP)

Version 7.05

Written by: Frank Kappe (fkappe@iicm.tu-graz.ac.at)
Gerald Pani (gpani@iicm.tu-graz.ac.at)
December 28, 1994

Contents

1 Introduction **3**
1.1 Status of this document . . . 3
1.2 Glossary . . . 3

2 References **5**

3 An Overview of the Protocol **6**
3.1 Opening and Closing the Connection . . . 6
3.2 Message Format . . . 7

4 Description of Messages **8**
4.1 Command 2: GETDOCBYANCHOR . . . 9
4.2 Command 3: GETCHILDCOLL . . . 9
4.3 Command 4: GETPARENT . . . 9
4.4 Command 5: GETCHILDDOCCOLL . . . 10
4.5 Command 7: GETOBJECT . . . 10
4.6 Command 8: GETANCHORS . . . 10
4.7 Command 9: GETOBJBYQUERY . . . 11
4.8 Command 10: GETOBJBYQUERYCOLL . . . 12
4.9 Command 11: OBJECTBYIDQUERY . . . 13
4.10 Command 12: GETTEXT . . . 14
4.11 Command 14: INSDOC . . . 14
4.12 Command 17: INSCOLL . . . 15
4.13 Command 19: GETSRCSBYDEST . . . 16

4.14 Command 22: MVCPDOCSCOLL 16
4.15 Command 23: MVCPCOLLSCOLL 17
4.16 Command 24: IDENTIFY 17
4.17 Command 25: READY 17
4.18 Command 26: COMMAND 18
4.19 Command 27: CHANGEOBJECT 19
4.20 Command 28: EDITTEXT 20
4.21 Command 29: GETANDLOCK 20
4.22 Command 30: UNLOCK 21
4.23 Command 31: INCOLLECTIONS 21
4.24 Command 32: INSERTOBJECT 21
4.25 Command 33: INCOLLSCLUSTER 22
4.26 Command 34: GETOBJBYFTQUERY 22
4.27 Command 35: GETOBJBYFTQUERYCOLL 23
4.28 Command 36: PIPEDOCUMENT 23
4.29 Command 37: DELETEOBJECT 24
4.30 Command 38: PUTDOCUMENT 25
4.31 Command 39: GETREMOTE 25
4.32 Command 40: GETREMOTECHILDREN 25
4.33 Command 41: PIPEREMOTE 26
4.34 Command 42: HG_BREAK 26
4.35 Command 43: HG_MAPID 26
4.36 Command 44: CHILDREN 26

5 Error Codes **27**

A An example C client **30**

1 Introduction

This specification describes the protocol between a Hyper-G client and the Hyper-G server (more specifically, the *link server* of Hyper-G). The protocol is described on a low level (i.e. the socket level), as this documentation is supposed to be used to create new Hyper-G clients (including gateways to other protocols) on arbitrary platforms and with arbitrary programming languages. It should be explicitly noted that the standard Hyper-G components (which are written in C++) usually do not bother about individual bytes sent over the network, but use a C++ class *HgStub* that is able to pass objects between clients and server. This explains why the protocol is not designed to be human-readable. It is, however, compatible with the *dispatch* library of the InterViews toolkit [5, 6].

In addition to the HyperG Client/Server protocol there is also a Server/Server protocol (used to maintain information base consistency across server boundaries), a Client/Client protocol (allows Hyper-G clients to pass messages and events to each other using the server), as well as protocols used for communication between the individual parts of the Hyper-G server (*hgserver, dbserver, ftserver, dcserver*; for an overview the reader is referred to the Hyper-G Architectural Design Document). These protocols are beyond the scope of this document.

1.1 Status of this document

The document you are currently looking at describes the Hyper-G Client/Server protocol version 7.05. An up-to-date version is available in hypertext form on the Hyper-G server of Graz University of Technology. If you prefer reading linear text on paper, a PostScript version is available by anonymous ftp from *ftp.iicm.tu-graz.ac.at* in file */pub/Hyper-G/papers/Protocol.ps.*

1.2 Glossary

This section is to define the meaning of a number of words that are used throughout this documentation.

- **anchor:** An anchor denotes a section of a document (perhaps all of it). For example, it may be a word or phrase in a text document, or a small area in a large picture. It is either the source or the destination of a link.
- **attribute:** Attributes are pieces of information attached to objects (e.g., as author, creation time, title, keywords, expiration time, type of object etc.). One can then use some of these to search the database for documents with certain attributes.
- **author:** In the Hyper-G system, the word 'author' refers to the user who entered a piece of information into the database–not necessarily the person who originally created the information. For example, in the case of a picture of the 'Mona Lisa', the author would be the person that inserted that picture into the system, rather than Leonardo da Vinci.
- **client:** The part of Hyper-G that displays documents and realizes the user interface.

- **cluster:** A specialized collection object (see below). Documents that are members of clusters should be visualized together (this is how multimedia objects are realized in Hyper-G).

- **collection:** A group of items, which may be either documents or collections. Can be compared to "directory" in a file system. This definition yields a "collection hierarchy", which looks like a tree, but with any number of "roots" and with the property that a collection may be a subcollection of several collections (- acyclic graph). For example, a biochemistry collection might be a subcollection of both a biology collection and a chemistry collection.

- **document:** A type of object that is intended to be displayed by the client. A document must be a member of at least one "parent collection", and may have a number of source and/or destination anchors attached. Hyper-G supports a number of document types, of which text documents are treated a bit special by the server (their contents are searchable), which is why in the protocol there are specialized functions that deal with text documents.

- **object:** Generic term for an entity stored in the Hyper-G database. Documents, links, anchors, and collections are examples of objects. An object has a set of attributes, each of which has one or more values. In particular, every object is assigned a unique object identifier.

- **link:** The essance of hypertext. A user may click on one endpoint of a link (the "source anchor") and jump to the other endpoint (the "destination anchor").

- **search:** All objects inserted into the Hyper-G database can automatic be searched for. The protocol supports basically two kinds of searches: Boolean searches based on attributes of objects, and full text searches that are defined for text documents only.

- **server:** The part of Hyper-G that stores the documents, links, etc.

2 References

[1] KAPPE F.: "Hyper-G: A Distributed Hypermedia System". In LEINER B. (editor), *Proc. INET '93, San Francisco, California*, pages DCC–1–DCC–9, Internet Society, August 1993.

[2] KAPPE F.: *Hyper-G Text Format (HTF)*. Technical Report, December 1993. Available by anonymous ftp from iicm.tu-graz.ac.at in file pub/Hyper-G/doc/HTF.ps.

[3] KAPPE F. and MAURER H.: "Hyper-G: A Large Universal Hypermedia System and Some Spin-Offs". *ACM Computer Graphics, experimental special online issue*, May 1993. Available by anonymous ftp from siggraph.org in directory publications/May_93_online/Kappe.Maurer.

[4] KAPPE F., PANI G., and SCHNABEL F.: "The Architecture of a Massively Distributed Hypermedia System". *Internet Research: Electronic Networking Applications and Policy*, 3(1):10–24, Spring 1993.

[5] LINTON M. A., CALDER P. R., and VLISSIDES J. M.: "The Design and Implementation of InterViews". In *Proc. USENIX C++ Workshop, Santa Fe, New Mexico*, November 1987.

[6] LINTON M. A., VLISSIDES J. M., and CALDER P. R.: "Composing User Interfaces with InterViews". *IEEE Computer*, 22(2):8–22, February 1989.