OpenIGTLink

Junichi Tokuda¹

¹Brigham and Women’s Hospital / Harvard Medical School,

Luis Ibanez², Csaba Csoma³, Patrick Cheng⁴,
Haiying Liu¹, Jack Blevins⁵, Junpei Arata⁶, Xenophon Papademetris⁷, Nobuhiko Hata¹

²Kitware Inc., ³Johns Hopkins University, ⁴Georgetown University,
⁵Acoustic Med Systems, ⁶Nagoya Institute of Technology, ⁷Yale University
Acknowledgement

• Support for Open IGT Link comes from
  – National Institutes of Health
    • National Center of Image Guided Therapy (PI: Jolesz)
    • National Alliance on Medical Image Computing (PI: Kikinis)
  – NSF ERC CISST (PI: Taylor)
  – NEDO, Japan
  – Dr. Kiyoyuki Chinzei, AIST, Japan
IGT system: academic perspective

- Hardware/software components
  - MR/CT/Ultrasound scanners
  - Position tracking devices
  - Robotic devices
  - Navigation software

- Data types exchanged among the IGT system
  - Images
  - Positions / Transforms
  - Commands
  - Software / hardware status, etc…
Related standards / works

• Communication standards used in medical area
  – Device connection
    • IEEE 1073 -- Medical Device Communication
    • ISO 11898 -- Controller Area Network (CAN)
  – Picture archiving and communication system
    • Digital imaging and communication in medicine (DICOM)

• Network communication framework for IGT
  – CORBA [Schorr-2000]
  – OpenTracker [von Spiczak]
Our goals

- Availability: research and commercial
- Simplicity: from embedded system to HPC
- Extensibility: variety of data types
- Reliability: data verification mechanism
Our solution: OpenIGTLink

- Community-oriented Development
  - The project was launched in US national level meeting 2008 (NA-MIC, all hands mtg, January 2008, Slat Lake City, UT).

- Platform-independent
  - Multi-platform C/C++ library for Windows/Linux/Mac

Application

C++ socket &
thread class

C++ message class

C message structure

Operating System
The OpenIGTLink Protocol

- Code snippet
  - Nine lines to send linear transform data

```cpp
igtl::ClientSocket::Pointer socket;
socket = igtl::ClientSocket::New();
socket->ConnectToServer("192.168.0.1", 18944);

igtl::TransformMessage::Pointer transMsg;
transMsg = igtl::TransformMessage::New();
transMsg->SetDeviceName("Tracker");
transMsg->SetMatrix(matrix);
transMsg->Pack();
socket->Send(transMsg->GetPackPointer(),
            transMsg->GetPackSize());
```
The OpenIGTLink Protocol

- Message-based protocol
  - No session / messages are independent
  - Allows defining new message types

```
Header
Version
Type Name
Device Name
Body Size
CRC
Time Stamp

Body
IMAGE Type Body
TRANSFORM Type Body
COMMAND Type Body
```

Body Size
Protocol

- Device name
  - Multi-channel or multi-device

- Body size
  - Allows receivers to skip data, even if type is unknown.

- CRC
  - Data integrity check in the receiving program

- Time stamp
  - Required in real-time application
Standard Data Types

- Position (tracking / device positioning)
  - Position: \(x\, y\, z\)
  - Orientation (quaternion): \(ox\, oy\, oz\, w\)

- Transformation (tracking / registration)
  - Position: \(tx\, ty\, tz\)
  - Scale: \(sx\, sy\, sz\)
  - Rotation: \(nx\, ny\, nz\)
  - Translation: \(x\, y\, z\)

\[
\begin{pmatrix}
  tx & ty & tz \\
  sx & sy & sz \\
  nx & ny & nz \\
  0 & 0 & 0 & 1
\end{pmatrix}
\]
Standard Data Types

- **Image**
  - 2D / 3D
  - Scalar / vector
  - Affine transform
  - Partial volume update
# Standard Data Type

## Status

<table>
<thead>
<tr>
<th>Code</th>
<th>Description</th>
<th>Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Invalid packet</td>
<td>10</td>
<td>Configuration error</td>
</tr>
<tr>
<td>1</td>
<td>OK</td>
<td>11</td>
<td>Resource error</td>
</tr>
<tr>
<td>2</td>
<td>Unknown error</td>
<td>12</td>
<td>Unknown instruction</td>
</tr>
<tr>
<td>3</td>
<td>Panic</td>
<td>13</td>
<td>Device not ready</td>
</tr>
<tr>
<td>4</td>
<td>Not found</td>
<td>14</td>
<td>Manual mode</td>
</tr>
<tr>
<td>5</td>
<td>Access denied</td>
<td>15</td>
<td>Device Disabled</td>
</tr>
<tr>
<td>6</td>
<td>Busy</td>
<td>16</td>
<td>Device not present</td>
</tr>
<tr>
<td>7</td>
<td>Time out</td>
<td>17</td>
<td>Unknown device version</td>
</tr>
<tr>
<td>8</td>
<td>Overflow</td>
<td>18</td>
<td>Hardware failure</td>
</tr>
<tr>
<td>9</td>
<td>Checksum error</td>
<td>19</td>
<td>Shutdown in progress</td>
</tr>
</tbody>
</table>
Applications

• Three highlighted scenarios

  1. Prototyping clinical system
  2. Research-commercial technological transition
  3. Clinical Research Bridging
1. Prototyping Clinical System

- Connect prototype hardware / software to the system
1. Prototyping Clinical System

• Surgical manipulator integration

Drs. H. Fujimoto and J. Arata, Nagoya Institute of Technology
1. Prototyping Clinical System

- Surgical manipulator integration

![Diagram of a clinical system with components such as 3D Motion Sensor (Optotrak), Navigation System (3D Slicer), Robot (PHANToM), and Markers.]
2. Research-Commercial Transition

- Replace research prototype with commercial product prototype

OpenIGTLink (Application Layer)

Component 1

Component 2

Research Prototype

Commercial Product

TCP/IP
2. Research-Commercial Transition

- MR-guided prostate robotic intervention
  - MRI-compatible needle placement robot [Fischer 2007]
2. Research-Commercial Transition

- MR-guided prostate robotic intervention
  - 3D Slicer: research prototype
3. Clinical Research Bridging

- Export clinical data from approved system (proprietary) to research software
3. Bridging to commercial system

• BrainLab – Slicer 3
  – *BrainLab*, commercial navigation system
  – *3D Slicer*, research platform
  – *BioImage Suite* (by Xenophon Papademetris, Yale University),
    bridging VVLink and OpenIGTLink protocols
Endowments

- GE Excite MRI
- JHU robots and encoders
- IGSTK -- NDI and Micron trackers
- Robin Medical EndoScout
- NIT robots (Intelligent SI Project)
- BrainLab via BioImage Suite (Papademetris)
- 3D Slicer
- Matlab interface
Summary

- **OpenIGTLink**
  - Open, simple, extensible and reliable
  - Slicer 3, MRI, tracking device, robot, etc.
  - Used in navigation and surgical robot projects

- **Communication protocol for IGT**
  - Prototyping clinical system
  - Research-commercial technological transition
  - Clinical research bridging
Future Work

• Complete toolbox for development
  – Testing tools
  – Monitoring / analysis tools

• Logging - “Blackbox”
  – Record all events in OR

• Hard real-time capability
  – Motion compensation in radiotherapy / FUS
For more information.....

google “open igt link”