

Standard Elevator Information Schema

The **Standard Elevator Information Schema** is a mechanism for communicating static information, such as configuration details, and events such as calls and car trips, between all manner of systems and users of passenger elevators. Information is sent between systems in the form of messages that can be interpreted both by computer systems human readers.

Based on XML

The idea of a standard language for communicating elevator information is not in itself new but recent advances in software technology, based on eXtensible Markup Language (XML), remove a number of technical barriers. By employing XML, it can benefit immediately from third-party development tools and software libraries that are already publicly available. It also conforms automatically to a widely accepted format for messaging. This leaves developers free to concentrate on those aspects of design that are the unique concern elevator engineers.

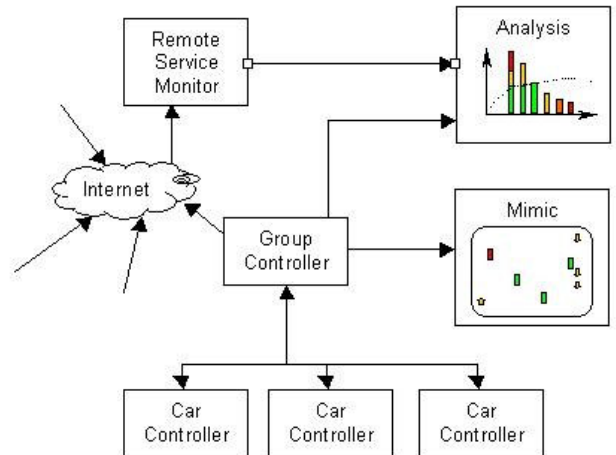
Scope of the Standard

The standard is designed to be universally applicable and is envisaged as being the medium for communicating a wide variety of messages within the elevator industry application domain. For example:

- Hardware events such as call button presses, door operation, etc that would drive car and group controllers.
- Time (and date) stamped logged data, such as would be communicated to mimic and monitoring equipment. This could equally be output by
 - operational controllers (both single car and group) or
 - from simulation programs.
- Analysis data, such as average system response time and a performance index for monitoring and fault diagnosis.
- Traffic and system configuration data for defining simulation runs.

(NB Although the above list talks about controllers, loggers, etc. there is no requirement for the information to be passed over communications link between two pieces of hardware. The standard is equally valid for communication between software

components that may well be running on the same computer hardware).



The above diagram is an example of a variety of application scenarios where each arrow is a flow of XML data conveying runtime information between the different components of a control and monitoring network.

Contents and Availability

The standard is supplied as a set of files including:

1. User Guide.
2. "[LiftInfo.XSD](#)" - The syntactic and semantic definition of the Standard.
3. Examples of the usage of the standard:

[Simulation Run](#)

[Group Controller configuration](#)

[Remote Performance Monitor](#)

[Event Data Log](#)

[In-Service Record](#)

The standard is freely available on the Internet to be used as widely as possible, without charge, by anyone who can benefit from it. The only condition of use is that the copyright of the author be acknowledged in any application where it is employed.

For further information and detailed definitions please refer to the Internet web-site of the Standard Elevator Information Schema at:

www.std4lift.info

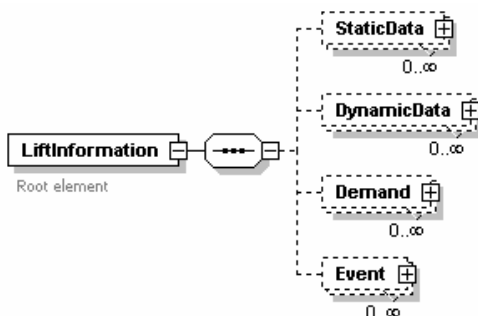
Standard Elevator Information Schema

Overview of the Schema

Although the schema of the Standard is quite precise about how messages are constructed and the information that they must contain, it nonetheless allows the inclusion of certain elements to be optional. So only those parts of a message that are necessary for the message to be valid and complete need be included. This flexibility makes the format very efficient.

The schema describes a wide variety of information to be communicated:

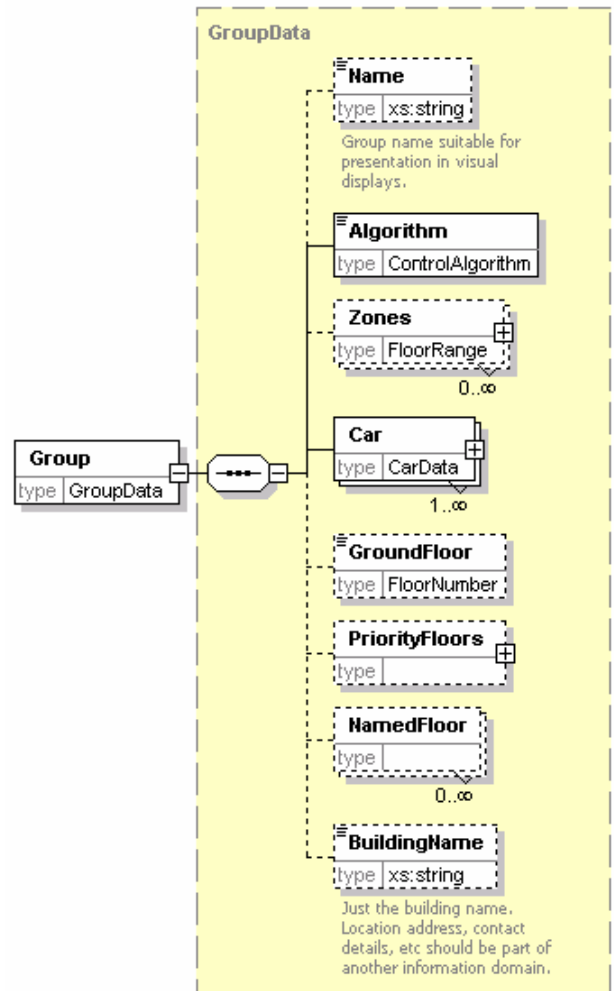
- **"Static"** types of information cover aspects of the elevator system that are known at the time of design e.g.:
 - Number of cars in a group
 - Floors in a building
 - Door operating times
 - etc
- **"Dynamic"** information is like a "snapshot" of the current state eg:
 - Current car position
 - Registered calls
 - Committed direction
 - etc
- **"Demand"** information records the traffic handling of the elevator system e.g.:
 - The profile of response times over a given period
 - etc
- **"Event"** information records e.g.:
 - A call is registered
 - A car stops
 - Car load changes
 - etc



The remainder of the schema merely defines the variety of types of information that make up these three "top-level" types, e.g.:

- StaticData consists of either a Group or Car element
- But a Group element will, itself, contain
 - one or more Car elements
 - along with information about
 - the group control Algorithm and
 - any Zones which are defined to improve service distribution
 - any special or priority floors in the building

and so on, breaking the information down into smaller and smaller units until the smallest, "atomic" level is reached.



NB. The Standard recognises that other standards already exist to define, for example, manufacture and construction information and postal addresses, etc and so such areas are specifically omitted to avoid conflict and duplication.