

---

# Orchestra

## An Introduction

---

### What is Orchestra?

Orchestra is a standard for creating machine-readable definitions of messaging protocols. It allows you to specify everything you need to describe your messaging system including:

- Which messages you have and what they are used for.
- Which fields (or groups of fields) go in each message, their formats and validation rules.
- A way of defining alternative message layouts or field values for different use cases, for example:
  - Message x looks like this in scenario A and like this in scenario B,
  - Field x supports these values in scenario A and those values in scenario B.
- Permitted messaging workflows (e.g., if I send x, I expect to receive y).
- The 'lower levels' of your messaging implementation – network details, session and encoding settings etc.

This is all done in a fully standardised and machine-readable structure, meaning you can exchange Orchestra files with your counterparties knowing they will be able to read them.

### Why do we need it?

Because creating and using manual documentation is inefficient, prone to error, and needs to be interpreted by humans for configuring or programming computer systems.

### What can you do with it?

At the very least you can generate messaging protocol specifications ("rules of engagement") in both human- and machine-readable forms. But you can in theory do so much more, for example:

- Documenting internal systems in a standardised and machine-readable way.
- Handling of message normalisations and mapping rules.
- Configuration of instances of a messaging system (e.g., session and encoding parameters, network addresses, connectivity details of your counterparties etc.).
- Automated certification testing of other peoples' (and indeed your own) rules of engagement against your infrastructure.
- Code generation to set and retrieve message content.



### Does it support the FIX Protocol?

Does it support the FIX Protocol? Yes - however, Orchestra can be used to model both FIX and non-FIX protocols. For example, we have modelled the NYSE Pillar's binary interface in Orchestra. It can also be used to describe data structures, for example as used by the US Office of Financial Research<sup>1</sup>, and even regulatory documentation.

### Does the FIX Trading Community use it?

Yes – the online specification for FIX Latest (the most recent version of the FIX Protocol) is built in Orchestra. We have also started to create our Recommended Practices documents from Orchestra<sup>2</sup> and plan to re-fit all our Recommended Practices documents to Orchestra.

### How do I get started?

FIX Trading Community members have access to a set of tools:

- **FIX Orchestra Server:** For creating rules of engagement documentation including its Orchestra representation.
- **Log2Orchestra:** Takes one or more FIX log files and creates an Orchestra representation of the messages in the file(s).
- **Playlist:** Allows the creation of an Orchestra representation by interactively selecting what is needed as a subset from an Orchestra reference file.

And for the general public, we have tools such as Tablature to convert markdown files to Orchestra files and vice versa.

There are also a number of converters from Orchestra to other representations such as the QuickFIX data dictionaries or schemas for Simple Binary Encoding and Avro (represented in JSON). This allows to use the power of Orchestra without having to change the encoding.

For more information, please go to [www.fixtrading.org/standards/fix-orchestra](http://www.fixtrading.org/standards/fix-orchestra) or contact us at [fix@fixtrading.org](mailto:fix@fixtrading.org).

---

<sup>1</sup> [www.fixtrading.org/fix-standards-now-part-of-the-ofr-financial-instrument-reference-database/](http://www.fixtrading.org/fix-standards-now-part-of-the-ofr-financial-instrument-reference-database/)

<sup>2</sup> [www.fixtrading.org/packages/recommended-practices-for-digital-asset-trading](http://www.fixtrading.org/packages/recommended-practices-for-digital-asset-trading)

