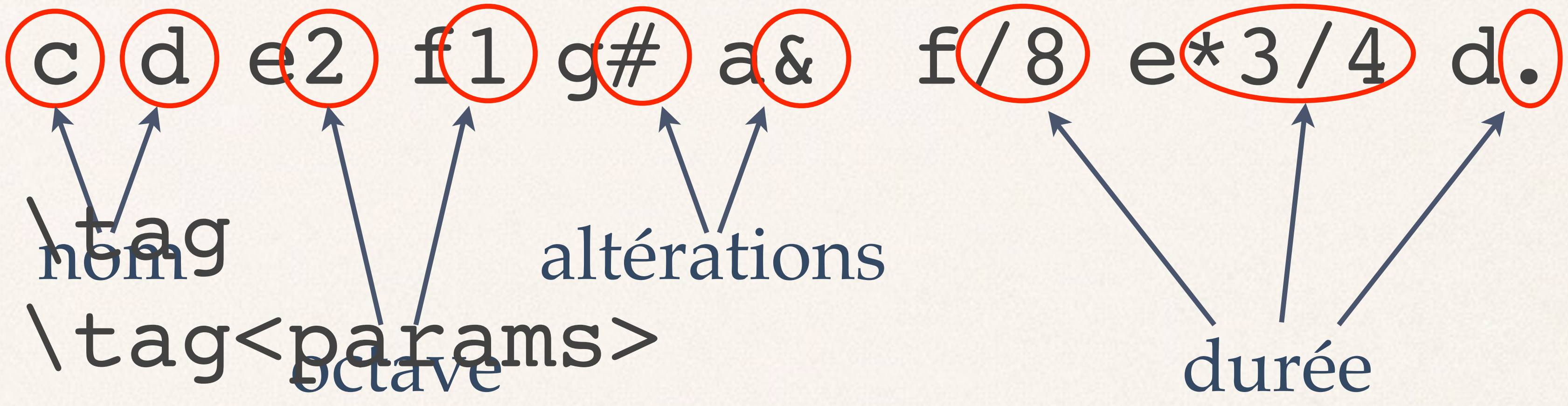
Déploiement des services du moteur de rendu de partitions GUIDO sur Internet.

D. Fober, M. Solomon

GRAME CENTRE NATIONAL DE CRÉATION MUSICALE

Guido Music Notation

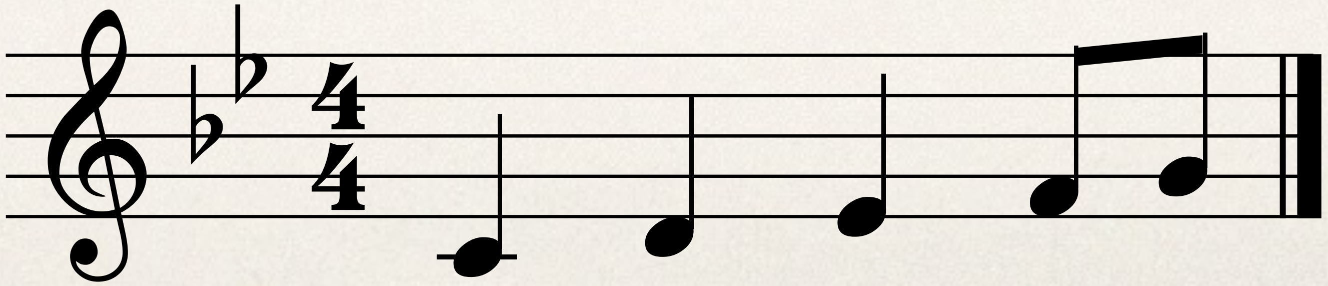
- Notes: `c d e2 f1 g# a& f/8 e*3/4 d.`



- Tags:

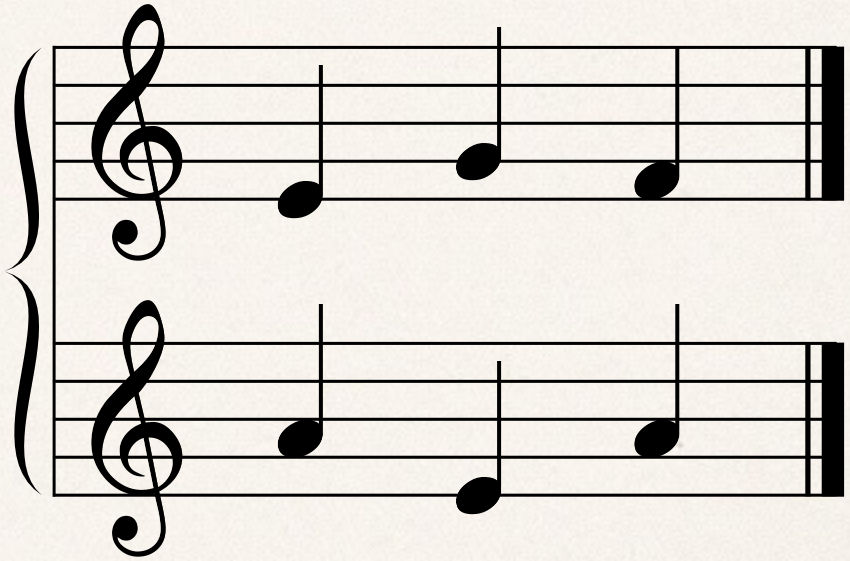
```
\tag
\tag<params>
\tag(notes-series)
\tag<params>(notes-series)
```

```
[ \key<-2> \meter<"4/4"> c d e& f/8 g ]
```



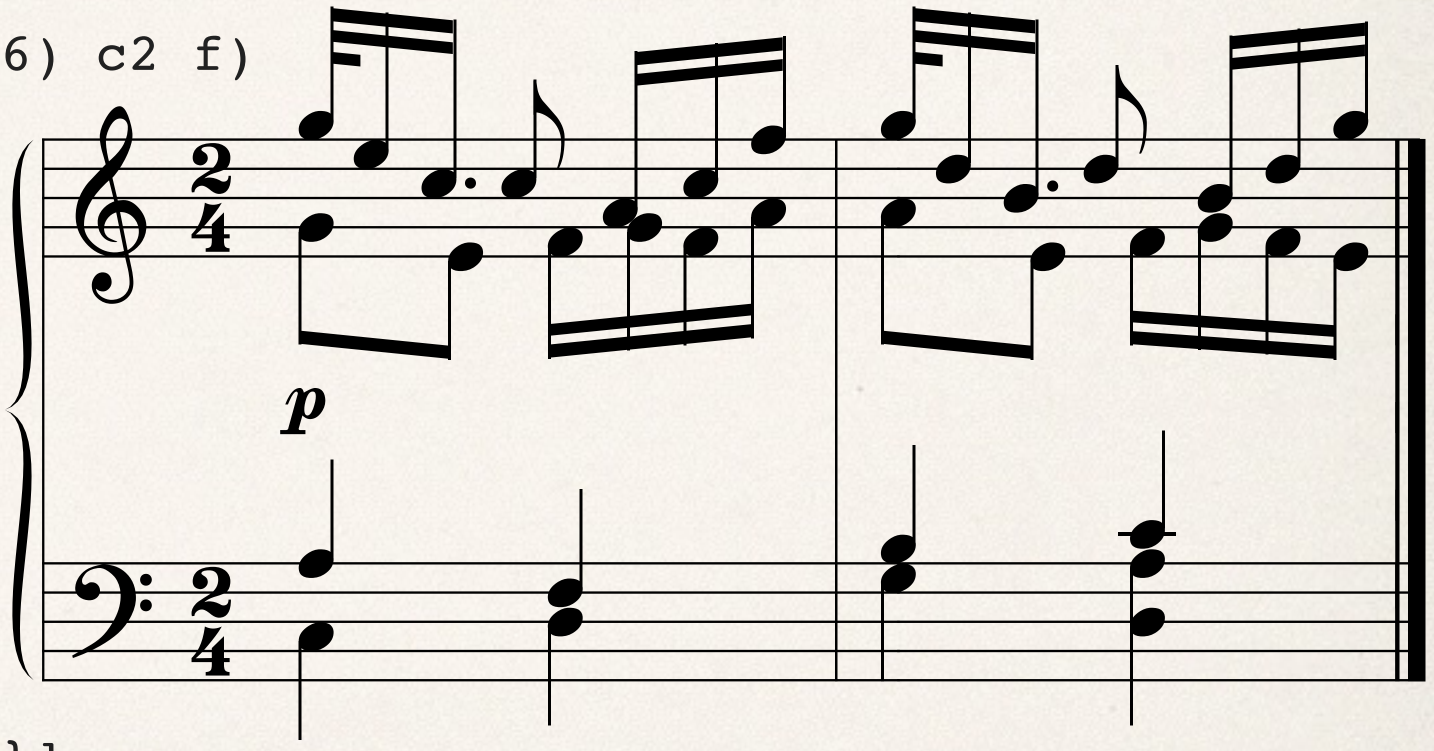
Guido Music Notation

```
{
 [ e g f ],
 [ a e a ]
}
```



Guido Music Notation

```
{
 [
  \barFormat<"system">
  \staff<1> \stemsUp \meter<"2/4">
  \intens<"p", dx=1hs,dy=-7hs>
  \beam(g2/32 e/16 c*3/32) c/8
  \beam(\noteFormat<dx=-0.9hs>(a1/16) c2 f)
  \beam(g/32 d/16 h1*3/32) d2/8
  \beam(h1/16 d2 g)
 ],
 [\staff<1>\stemsDown g1/8 e
  f/16 \noteFormat<dx=0.8hs>(g)
  f a a/8 e f/16 g f e
 ],
 [\staff<2> \meter<"2/4">
  \stemsUp a0 f h c1
 ],
 [\staff<2> \stemsDown c0 d g {d, a}]
}
```



GUIDO Engine

- Compiles Guido Music Notation (GMN) markup language.
- Interprets this into an abstract representation.
- Translates abstract representation into a graphical representation.
- Draws representation onto an abstract device.

GUIDO Engine

- Provides a C/C++ API:
  - to activate the compilation services:

    `GuidoParseFile, GuidoParseString, GuidoAR2GR…`
  - to draw the score onto a device:

    `GuidoOnDraw,GuidoSVGExport…`
  - to query the scores,

    `GuidoCountVoices, GuidoGetPageCount, GuidoDuration…`
  - to get the graphic to time relationship

    `GuidoGetVoiceMap, GuidoGetStaffMap…`

State of the Art

- Notation editors (GUI)
- Score sharing software
- JIT compilation services

Problem

- Lack of low-latency alternative to create musical notation.
- Lack of an API that allows for web application building.

Solution

- Provide a service that quickly compiles music notation representations into several encoded forms (PNG, SVG, DSL).
- Provide a web API via a server-client architecture.

REpresentational State Transfer

Architectural style

- Client-server model
- Stateless
- Cacheable
- Uniform interface
- Layering
- Code on demand (optional)

REST design practices

- Standard HTTP methods (GET, PUT, POST or DELETE)
- Transferring metadata along with data
- Human readable, hierarchical URIs
- URI points to idea, not necessarily to permanent object

GUIDO HTTPD Server

- Uses GNU libmicrohttpd to handle POST, GET, and DELETE requests (uniform interface)
- *almost* stateless - use of SHA1 key as resource identifiers prevent too-long URIs

GUIDO Web Api

- Content as SHA-1 key
  `[ g e c ] => da8e1434e155e8a20f328de7d7ea5874d149f5ee`

- Function as URI segment
  `http://server/da8e1434e155e8a20f328de7d7ea5874d149f5ee/`***`duration`***

- Arguments to functions and options as key-value pairs
  `…/da8e1434e155e8a20f328de7d7ea5874d149f5ee`***`?format=svg`***

- Return values as standard MIME types
  `image/png , application/json, image/svg+xml,…`

guidoeditor.grame.fr

- Example uses server-side layout calculations and client side drawing to avoid lengthy transfers.
- Also possible to do server-side drawing.
- Potential for creating fast embedded scores as well as simple editing applications.

Conclusion

GUIDO has the potential to be the open-source alternative for fast music representation and could be useful for many of the applications used today, including online services deployment.

Questions

- guidolib.sf.net
- guidoeditor.grame.fr
- guidoservice.grame.fr
- guido.grame.fr