

El futur del programa

Stable i Developer releases

L'autor del bumblebee havia escrit: [The future of Bumblebee](#)

On també s'explica que:

Stable releases: Bumblebee is a system on which hundreds of people rely. Once a release is marked stable, it really should be that. Moreover, from then on, that series of releases should also be stable with only bugfixes and occasionally small, self-contained new features being added (in a way that won't break previous installations). Following the common open-source release numbering scheme, stable releases have an even number as their minor version number, for example 1.0.z, 1.2.z, etc. (download)

Developer releases: The only way to develop and test new features is by releasing code... "release early, release often". While the developer releases are marked as "unstable", we do believe that they are data-safe; they shouldn't corrupt your existing data and should be safe to use... there might just be a few pretty ugly bugs in them (which we hope you will report!). Of course, you always keep your data backed up, don't you. Developer releases have an odd number as their minor version number, such as 1.1.z, 1.3.z, etc. (download)

Nosaltres seguirem aquesta filosofia a l'hora d'etiquetar les diferents versions.

Plans futurs

Nosaltres seguim algunes coses que va escriure l'autor i també afegim coses noves. Aquesta taula és una barreja entre el que va escriure l'autor i el que em fet o tenim pensat fer i el que em escrit nosaltres. S'accepten suggerències.

1.4

- * Cua de rutina
- * Utilitzar ajax (per exemple, quant creem un usuari i seleccionem el seu grup, que es carregui la llista de projectes lligats a aquell grup)
- * Idioma interactiu (internationalisation)
- * Reserves amb confirmació
- * Millora de la gestió de consumibles
- * improve timeslot config
 - validation on the timeslotrule input e.g. making sure every hour is accounted for
 - better UI for setting timeslots

1.2

- * Reports and Lists pels caps de grup (per grup i projecte)
- * La connexió entre usuaris i grups es fa ara a través de la nova taula `usersgroups` de la base de dades
- * Nou tipus d'usuari (`Main_Researcher` o `Supervisor` o Cap de grup). D'això deriva una nova taula a la base de dades; `statususers`
- * L'administrador i l'administrador d'un instrument poden fer una reserva per qualsevol projecte
- * internationalisation (not interactive yet, just can change language in `bumblebee.ini` for all users)
- * PHP5 support
- * more reporting functions:
 - summary of bookings for each user

Petits canvis previstos que anem apuntant (a partir de la versió 1.1.7)

- A la taula `statususers` el camp `'deleted'` no hi pinta res, però si no li posem, actualment no funciona (no podem ni editar ni crear usuaris). S'ha de mirar perquè i com solucionar-ho. **DONE 1.3.0** Ja no és necessari el camp `deleted` a la taula `statususers`.
- A la pantalla principal d'Instruments, si l'usuari és administrador de l'instrument, indicar-ho. **DONE 1.3.0** Si ets administrador d'instrument, els instruments dels quals ets admin, et surten en un altre requadre.
- A `MyAccount`, si l'usuari és administrador de l'instrument, indicar-ho. **DONE 1.1.7** Amb un checkbox no editable, ho indiquem.
- Al crear usuari, el camp `password` ha de ser obligatori. **CANCELED:** És millor tal com està sinó hauríem d'activar que es pogués accedir al valor del password i és més segur no poder-ho fer. S'ha de procurar crear un usuari amb password, si no té password, no podrà accedir al programa.
- Al crear un usuari, s'ha de comprovar si l'username ja existeix, sinó la base de dades petja. **DONE 1.3.0** Si l'error és "fatal" està bé que peti, però en comptes de només ensenyar l'error: "*Ooops. Something went very wrong...*" ara també ensenya el motiu de l'error (tant si és "fatal" com si no), exemple: "*Duplicate entry 'admin' for key 'username'*". Si volem veure la consulta o consultes que són incorrectes, hem d'activar `VerboseSQL=1`.
- L'administrador d'instrument, al fer `masquerade`, només li haurien de sortir el llistat d'usuaris amb permís al instrument del qual és administrador. **DONE 1.3.0**
- Opció per esborrar l'usuari definitivament (de la base de dades). **CANCELED:** També hauríem d'esborrar les seves reserves, unions amb grups i projectes... I volem l'historial per si el necessitem algun dia.
- Mirar si realment necessitem la variable `isadmin` tinguent el camp `status=1`.
- Canviar el codi per enviar mails globals o de reserves cancel·lades (`My account`)
- Em de pensar com utilitzar el `isdefault` de la taula `usergroups`.
- Formulari de crear un booking dinàmic (per la selecció de projectes meus o de projectes d'altres).
- Preguntar: per què grup Antoni Morros `deleted = 1` i en canvi el projecte (com tots els projectes) `deleted = 0`?

Cua de rutina - IconNMR

Per ara deixo aquí la informació que he recopilat sobre la programació externa d'experiments a l'IconNMR (cua de rutina),

Per fer

- Decidir quins experiments s'ofereixen a la cua de rutina. Consultar els experiments a les carpetes C:\Bruker\TOPSPIN\exp\stan\nmr\par.iconnmr i C:\Bruker\TOPSPIN\exp\stan\nmr\par.SERMN de l'espectròmetre Avance DPX-250 ROBOT
- Decidir quins paràmetres d'adquisició deixem que pugui modificar l'usuari. Alternativament, es podria definir un catàleg més ampli d'experiments, per exemple, amb diferents *sw*, *ns*, *d1*, etcètera.
- Decidir quins solvents s'ofereixen a la cua de rutina.
- Decidir com s'indica si la mostra ja ve preparada per l'usuari o si l'hem de preparar nosaltres. Cal guardar aquesta informació al programa de reserves?
- Decidir com s'indica si cal processar i representar els experiments. Cal guardar aquesta informació al programa de reserves?
- Redactar unes notes breus sobre com exportar els fitxers i importar-los a l'IconNMR.
- Què fem amb les opcions *SUBMIT* i *NO_SUBMIT*?
- Decidir el títol dels experiments, per exemple: `<data> <hora> - <sample name> \n <experiment> - <solvent> - <parameter set> [(modif)]` on *(modif)* només s'afegeix si s'ha modificat algun valor per defecte de l'experiment (parameter set) triat.

Fitxers per programació d'experiments

Aquest és un exemple de fitxer per programar experiments a l'IconNMR,

```
# iconnmr external setup file
#
# dual experiment test file - default parameters modified
#
USER SeRMN
HOLDER 33
NAME ext-set-test-003
SOLVENT acetone
#
# begin of 1st experiment
#
EXPNO 1
NO_SUBMIT
PARAMETERS sw,20,olp,0,ns,512
EXPERIMENT teo_proto
TITLE ext-set-test-004\n 1D-1H-experiment
#
# end of 1st experiment
```

```
#  
# begin of 2nd experiment  
#  
EXPNO 2  
NO_SUBMIT  
PARAMETERS ns,32k  
EXPERIMENT teo_carboni  
TITLE ext-set-test-00\n 1D-13C-experiment  
#  
# end of 2nd experiment  
#  
# end of holder definition  
END
```

Altres fitxers d'exemple emprats a les proves. **COMPTE!** abans d'importar-los cal substituir les extensions .ext per .001, .002,... d'acord amb el número al final del nom del fitxer -001, -002,...)

- [iconnmr-external-setup-001.txt](#)
- [iconnmr-external-setup-002.txt](#)
- [iconnmr-external-setup-003.txt](#)
- [iconnmr-external-setup-004.txt](#)
- [iconnmr-external-setup-005.txt](#)

Notes sobre l'ús de fitxers externs per programar experiments

Aquestes són les notes que vaig prendre mentre feia proves amb diferents fitxers. Caldrà tenir-les present a l'hora d'introduir la informació a la base de dades del Bumblebee, i quan es generin els fitxers.

- La importació dels fitxers és gairebé immediata. Tan bon punt es copien a la carpeta C:\Bruker\TOPSPIN\prog\tmp el programa els importa i mostra a la pantalla de l'IconNMR.
- Els fitxers importats es mouen a la carpeta C:\Bruker\TOPSPIN\prog\tmp\save, de forma que, en cas de necessitat, es podrien recuperar, editar (si calgués), i moure de nou a la carpeta C:\Bruker\TOPSPIN\prog\tmp.
- Després del paràmetre EXPERIMENT només pot anar TITLE, qualsevol altre paràmetre es descarta. Així doncs, cal respectar l'ordre de definició dels paràmetres que es mostra al fitxer d'exemple anterior.
- El nom de l'usuari ha de ser exactament un dels definits a l'IconNMR, i les majúscules i minúscules no són equivalents (case sensitive!)
- El nom del solvent ha de ser un dels noms reconeguts pel TopSpin/IconNMR
- Si el nom de la mostra es repeteix encara que sigui en un altre holder, els EXPNO s'incrementaran perquè no coincideixin amb els EXPNO de la mostra d'igual nom en un holder diferent.
- No sé quin és el límit per TITLE, però accepta frases tan llargues com

```
TITLE <data - hora> - ext-set-test-00\n 1D-13C - acetone - teo_carboni  
(modif)
```

sense problemes.

- Si per error s'assigna el mateix HOLDER i EXPNO a una segona mostra, els experiments es programen al HOLDER indicat, però els EXPNO s'incrementen per no coincidir amb els ja definits. Malauradament, com que només es pot posar una mostra per holder, el que passarà és que a l'IconNMR haurem de moure aquests experiments al holder correcte.
- Si s'envien nous experiments a un holder amb experiments definits, els nous experiments s'afegeixen darrera dels existents, amb l'EXPNO incrementat per respectar els existents. Però si s'afegeix el paràmetre DELETE darrera del paràmetre HOLDER, els experiments existents s'esborraran abans d'afegir els nous, sense tenir en compte l'estat dels experiments (submitted, available, etc.).

 **Fix Me!** Vaig fer varies captures de pantalla del programa IconNMR després d'importar diferents fitxers. Aquestes captures il·lustren els errors i problemes que es poden produir. Quan tingui una estona les afegiré i comentaré.

From:
<https://sermn.uab.cat/wiki/> - **SeRMN Wiki**

Permanent link:
https://sermn.uab.cat/wiki/doku.php?id=sermn_wiki:userpages:marta:reserves_future&rev=1329816011

Last update: **2012/02/21 10:20**

