Sending small files

Problem

Sending files has a substantial constant overhead, and requires multiple online presenses from both sides, with additional connection handshake. For large files this overhead is justified, as otherwise files would consume queue quota, but for small files it makes sending files slow.

Solution

Send small files in the same connection. There can be two modes of sending files - the one that requires explicit acceptance, but after the acceptance the file will be delivered inline, without creating a new conneciton. Another, when the file will be sent straight after the message - that would require preliminary agreement, per contact - this mode can be useful for small voice messages and gifs.

Design

1. Add optional `fileInline :: Maybe FileInlineMode` property to `FileInvitation`:

haskell data FileInlineMode = FIInvitation -- recepient must accept | FIChunks -- file is sent after the message without acceptance

1. Add `XFileAcptInline SharedMsgId String` message to accept inline files, this can only be sent in case inline mode is offered, so the sender would support it:

{ "properties": { "type": {"enum": ["x.file.acpt.inline"]}, "msgId": {"ref": "base64url"}, "params": { "properties": { "msgId": {"ref": "base64url"}, "fileName": {"type": "string"} } } } }

1. Add `XFileChunks` message that have to be sent in front of the sequence of chunks (sent only in `FIInvitation` mode):

{ "properties": { "type": {"enum": ["x.file.chunks"]}, "msgId": {"ref": "base64url"}, "params": { "properties": { "msgId": {"ref": "base64url"}, "fileName": {"type": "string"} } } } }

1. Support file chunks in the main connection if the previous message was `XFileChunks` or `FileInvitation` in `FIChunks` mode.