

## NEGOTIATOR - PEER PROTOCOL :

- \* Her eş büyük sisteme dahil olabilmek için öncelikle NEGOTIATOR'a bağlanacaktır.
  - \* NEGOTIATOR bağlanan eşe diğer eşlerin bağlantı bilgilerini iletecektir --> CONNECT\_POINT  
(Yani her bağlanan eş, sunucuya kaydolup kendi bağlantı adresini ve portunu sağlamalıdır)
  - \* Belirli zaman aralıklarıyla (UPDATE\_INTERVAL) eşlere kendisine bağlı olan eşlerin bağlantı adresi listesini  
iletmelidir veya her eş belirli zamanlarda bu bilgiyi istemelidir.
  - \* Sunucu üzerinden görüntü verisi alışverişi yapılmayacaktır.
  - \* Yeni bir eş bağlandığında ve kendisini kaydettiğinde (REGISTER), sunucunun bir eş gibi o eşe bağlanmaya çalışıp  
[PEER IP, PEER PORT] ikisilinde bir problem olup olmadığını kontrol etmesi gerekir.
  - \* Merkez sunucu belirli zaman aralıklarında kendi üzerindeki bağlantı adres listesini  
(CONNECT POINT LIST) güncelleme çalışması yapacaktır.
  - \* sunucu-eş arasındaki protokol soru-cevap (senkron) şeklinde olacaktır.
  - \*\*\*\*\* sunucu eşlere sorgu yapmadığı sürece bir şey söylemeyecektir.
  - \* dolayısıyla bu bağlantının eşlerde asenkron veri beklemesi gerekmemektedir.
  - \* NEGOTIATOR-PEER bağlantısının amacı ise PEER buluşturma olacaktır.
- negociator --> amaci tamamen baglanti bilgisinin paylasimi

## NEGOTIATOR - PEER PROTOCOL :

NEGOCIATOR-PEER protokol sayesinde eşler sunucuya kendilerini kayıt ederler. Yine bu protokol sayesinde kendilerini  
authenticate ederler sunucuyla. Bunun dışında bu protokol sayesinde eşler kendilerine bağlı olan eşlerin bağlantı adresleri  
bilgisini elde ederler. NEGOTIATOR veritabanının üzerinden bilgileri ancak bu protokol sayesinde edinebilirler.

NEGOTIATOR-PEER protokol kesinlikle soru-cevap şeklindedir (senkron). Burada sadece eşler NEGOTIATOR a istekte bulunabilir ve

NEGOCIATOR bu istekler doğrultusunda cevap verecektir. (tek cevap mı daha fazla mı ??? )

SUNUCU-EŞ sorguları :

PRPULSE :

Peer Pulse dediğimiz bu mesaj PEER\_CLIENT 'tan NEGOCIATOR a yollanan bir bildirimdir. Bu bildirimin amacı bu mesajı yollayan

eşin halihazırda çevrimiçi olduğunu ve işlem görmek istediğini iletir. Belirli bir süre zarfı içerisinde eşlerimizin her

birinden bu mesajın sunucuya gelmesi beklenmektedir. ( bu süreye şimdilik 30 saniye diyelim ). Eğer 120 saniye boyunca bu

bildirim sunucuya gelmezse sunucu bu eş çevrimdışı kabul eder ve listesinden bu eş çıkarır. Her PRPULSE mesajı için sunucu

NEGOK ( ok mesajı ) veya NEGERR (hata mesajı) cevabı vermektedir.

Girilen komut : PRPULSE

Request Fields ?

Beklenen cevap : NEGOK veya NEGERR. NEGERR bağlantının başarısız olması durumunda yollanacaktır. NEGOK mesajında ise NEGOK un

yanında request yapan eşin ismi (ID), IP si ve durumu (statusu) verilecektir. ( <komut> , <user\_name> , <user\_IP> )

örnek : NEGOK , sakir , 212.36.37.38 . Bu bir tuple şeklinde gönderilebilir.

PRLOGIN : (REGISTER)

PRLOGIN eşin büyük sisteme dahil olabilmek için NEGOCIATOR a request yaptığı bir komuttur. Bu request sayesinde aday olan eş

sisteme girebilir veya giremeyebilir. Sunucu eşini kabul etmeden önce mesajın formatına bakar. Mesaj formatı şöyle olmalıdır :

(<PRLOGIN>, <user\_name>, <user\_IP>). Sunucu kendi CONNECT\_POINT\_LIST'ine bakacaktır ve eğer login olmak isteyen eş bu listede

mevcut değilse ve yollanan mesaj doğru formatta atılmışsa sunucu bu yeni eşini sisteme dahil edecektir.

Karışılığında yine NEGOK veya NEGERR mesajı yollayacaktır. NEGOK mesajı döndürdüğünde , NEGOK un yanına Merhaba ve kullanıcı

ismini de yazacaktır. NEGERR durumunda ise neden eşini kabul etmediğini yazacaktır NEGERR in yanında.

format : ( <NEGOK> , <Merhaba> , <user\_name> )

( <NEGERR>, <reason> )

örnek : ( NEGOK , Merhaba , veli\_34 )

( NEGERR , böyle bir kullanıcı zaten sistemde mevcut )

( NEGERR , Yanlış format )

PRCLOSE : Client Offline Message

PRCLOSE ile eş sistemden çıkmak istediğini belirtir. Bu mesajı alan sunucu yine diğer durumlarda olduğu gibi ya NEGOK ya da

NEGERR ile cevap verecektir. Mesaj yanlış formatta atıldıysa, sunucu buna NEGERR ile cevap verecektir aksi halde NEGOK ile cevap

verecektir.

format : ( <NEGOK> , <Hoşçakal>, <user\_name> )

( <NEGERR> , <reason> )

örnek : ( NEGOK , Hoşçakal , ahmet\_dursun )

( NEGERR , yanlış format )